

General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

BOEING

Solid Rocket Booster
Performance Evaluation Model
VOLUME I — ENGINEERING DESCRIPTION

(NASA-CR-150335) SOLID ROCKET BOOSTER
PERFORMANCE EVALUATION MODEL. VOLUME 1:
ENGINEERING DESCRIPTION (Boeing Aerospace
Co., Huntsville, Ala.) 351 p HC A16/MF A01

N77-27181

Unclas
CSCL 21H G3/20 36835



CONTRACT NASS-29643

DCN 1-2-50-23786
SOLID ROCKET BOOSTER PERFORMANCE EVALUATION MODEL

Solid Rocket Booster Performance Evaluation Model **VOLUME I — ENGINEERING DESCRIPTION**

PREPARED BY

BOEING AEROSPACE COMPANY
(A DIVISION OF THE BOEING COMPANY)
HUNTSVILLE, ALABAMA

C. R. CARTER — PROJECT SUPERVISOR

SEPTEMBER 7, 1974

PREPARED FOR

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
GEORGE C. MARSHALL SPACE FLIGHT CENTER
HUNTSVILLE, ALABAMA

CONTENTS

<u>SECTION</u>		<u>PAGE</u>
	CONTENTS	
	LIST OF FIGURES	
	LIST OF TABLES	
	ACKNOWLEDGEMENTS	
1.0	INTRODUCTION	1-1
2.0	EXECUTIVE MODULE	2-1
2.1	EXECUTIVE MODULE ENGINEERING DESCRIPTION	2-1
2.2	SUBROUTINE INIT1 ENGINEERING DESCRIPTION	2-3
2.3	SUBROUTINE PRESET ENGINEERING DESCRIPTION	2-3
3.0	CENTRALIZED INPUT MODULE	3-1
3.1	SUBROUTINE INPUT ENGINEERING DESCRIPTION	3-1
3.2	SUBROUTINE GETDAT ENGINEERING DESCRIPTION	3-2
4.0	BATES ISP MODULE (BATES)	4-1
4.1	BATES ISP MODULE ENGINEERING DESCRIPTION	4-1
4.1.1	BATES Test Motor Performance	4-1
4.1.2	End Item Motor Performance	4-4
4.2	SUBROUTINE ITIDNZ AND DATLOC (IDNOZ) ENGINEERING DESCRIPTION	4-8
4.3	SUBROUTINE LEWIS AND LEWIT ENGINEERING DESCRIPTION	4-8
4.4	SUBROUTINE IDNOZL ENGINEERING DESCRIPTION	4-9
4.5	SUBROUTINE LESSQ ENGINEERING DESCRIPTION	4-9
4.6	BATES ISP MODULE LIMITATIONS	4-10
5.0	ISP SCALING MODULE (SISCAL)	5-1
5.1	ISP SCALING MODULE ENGINEERING DESCRIPTION	5-1
6.0	CONTRACTOR DATA ISP MODULE (CDSI)	6-1
6.1	CONTRACTOR DATA ISP MODULE ENGINEERING DESCRIPTION	6-1
6.2	CONTRACTOR DATA ISP MODULE LIMITATIONS	6-5
7.0	INTERNAL BALLISTICS MODULE (IBM)	7-1
7.1	INTERNAL BALLISTICS MODULE ENGINEERING DESCRIPTION	7-1
7.2	INTERNAL BALLISTICS MODULE LIMITATIONS	7-1
8.0	THRUST SCALING MODULE (FSCAL)	8-1
8.1	THRUST SCALING MODULE ENGINEERING DESCRIPTION	8-1
8.2	THRUST SCALING MODULE LIMITATIONS	8-3
9.0	"INERT" MASS MODULE (INERT)	9-1
9.1	"INERT" MASS MODULE ENGINEERING LIMITATIONS	9-1

CONTENTS (Continued)

<u>SECTION</u>		<u>PAGE</u>
10.0	RECONSTRUCTION MODULE (RECON)	10-1
10.1	RECONSTRUCTION MODULE ENGINEERING DESCRIPTION	10-1
10.1.1	Reconstruction Calculations in IBM Subroutine MNCHN4	10-5
10.2	SUBROUTINE BIAS ENGINEERING DESCRIPTION	10-6
10.3	FUNCTION PADJ(TIME) ENGINEERING DESCRIPTION	10-6
10.4	RECONSTRUCTION MODULE LIMITATIONS	10-6
11.0	DISPERSION MODULE	11-1
11.1	DISPERSION MODULE ENGINEERING DESCRIPTION	11-1
12.0	OUTPUT MODULE	12-1
12.1	OUTPUT MODULE ENGINEERING DESCRIPTION	12-1
13.0	NOZZLE SUBMERGENCE AND CONTOUR EFFECTS MODULE (NSCE)	13-1
13.1	NOZZLE SUBMERGENCE AND CONTOUR EFFECTS MODULE ENGINEERING DESCRIPTION	13-3
13.1.1	Contour Effects Simulation (IOPT=1)	13-3
13.1.2	Nozzle Submergence Effects Simulation (IOPT=2 and 3)	13-5
13.1.2.1	Steady State Solution of Submerged Nozzle Gas Dynamics (IOPT=2)	13-10
13.1.2.2	Non-Steady State Solution of Submerged Nozzle Gas Dynamics (IOPT=3)	13-12
13.2	SUBROUTINE MACH (ARINPT, GAMA, M) ENGINEERING DESCRIPTION	13-15
13.3	SUBROUTINE SUBSON (ICHOKE, IOPT) ENGINEERING DESCRIPTION	13-16
13.4	NOZZLE SUBMERGENCE AND CONTOUR EFFECTS MODULE LIMITATIONS	13-17
14.0	REFERENCES	14-1
	APPENDIX:	A-1
	SOLID PROPELLANT ROCKET MOTOR INTERNAL BALLISTICS COMPUTER PROGRAM - PROGRAM MANUAL REVISION	

LIST OF FIGURES

<u>FIGURE</u>		<u>PAGE</u>
1-1	MODULARIZED SRB-II PROGRAM FLOWCHART	1-2
2-1	OPTIONS TREE FOR THE PREDICTION MODULE OF SRB-II	2-4
2-2	OPTIONS TREE FOR THE RECONSTRUCTION MODULE OF SRB-II	2-6
13-1	INTERNAL FLOW WITH A SUBMERGED NOZZLE	13-2
13-2	DEFINITIONS OF THE ANGLES α AND θ_{ex} FOR A CONTOUR NOZZLE	13-2
13-3	ASSUMED AFT-END CONFIGURATION WITH A SUBMERGED NOZZLE	13-6
13-4	DEFINITIONS FOR MASS ADDITION REGION A	13-7
13-5	DEFINITIONS FOR MASS ADDITION REGION B	13-8
13-6	MASTER ITERATION SCHEME FOR SUBMERGED NOZZLE GAS DYNAMIC SOLUTION	13-9

LIST OF TABLES

<u>TABLE</u>		<u>PAGE</u>
2-I	SUBROUTINES AND MODULES CALLED BY THE EXECUTIVE MODULE	2-1
4-I	BATES ISP MODULE SUBROUTINES	4-1
5-I	ISP SCALING MODULE SUBROUTINES	5-1
6-I	CONTRACTOR DATA ISP MODULE SUBROUTINES	6-1
8-I	THRUST SCALING MODULE SUBROUTINES	8-1
10-I	SUBROUTINES AND MODULES CALLED BY THE RECONSTRUCTION MODULE	10-1
11-I	SRB DISPERSION VARIABLES	11-1
12-I	OUTPUT MODULE SUBROUTINES	12-1
13-I	NSCE MODULE SUBROUTINES	13-1
A-1	PAGES TO BE DELETED	A-3
A-2	PAGES TO BE REVISED	A-3

D256-10020-1

ACKNOWLEDGEMENTS

Contributions to the development, programming and documentation of the Solid Rocket Booster Performance Evaluation Model were made by the following personnel:

John R. Colson

Emory E. Lynn

James S. Richards

Charlotte G. Wiser

Any questions concerning this document should be directed to the responsible Project Supervisor.

Charles R. Carter

JD-13
895-2650

SECTION 1

1.0 INTRODUCTION

The Space Shuttle Solid Rocket Booster Performance Evaluation Model (SRB-II) is made up of analytical and functional simulation techniques linked together so that a single pass through the model will predict the performance of the propulsion elements of a Space Shuttle Solid Rocket Booster. The flowchart given in Figure 1-1 shows each of the SRB-II modules and the various user controlled options. The available options allow the user to predict static test performance, predict nominal and off nominal flight performance, and reconstruct actual flight and static test performance. Options selected by the user are dependent on the data available. These can include data derived from theoretical analysis, small scale motor test data, large motor test data and motor configuration data. The user has several options for output format that include print, cards, tape and plots. Output includes all major performance parameters (Isp, thrust, flowrate, mass accounting and operating pressures) as a function of time as well as calculated single point performance data.

This document contains Volume I of a set of four Volumes which describe SRB-II. Volume I contains the engineering description, Volume II contains the users manual, Volume III contains the sample case and Volume IV contains the program listing.

The engineering description of SRB-II contained in Volume I describes the basic approach used to develop each module. Sufficient detail is given to allow the reader an understanding of the engineering and programming fundamentals used, the function of each module, and the limitations of each module. Program limitations are dependent on the program option selected by the user. Limitations or restrictions that apply to a particular module will be a part of the Module Engineering Description. The three existing computer programs incorporated into SRB-II (Boeing Internal Ballistic Computer Program, Lewis Thermochemical Computer Program, and One-Dimensional Two-Phase Flow Loss Computer Program) are not described in detail except as how they are used. The engineering description of these programs may be found in the References.

In each of the following module engineering descriptions, the basic equations used in SRB-II will appear in the identical Fortran format used in SRB-II. It is assumed that the reader is familiar with the standard fortran notation and no attempt will be made to define the notation in this document. Module engineering descriptions in Volume I follow the format of the Module program listings given in Volume IV.

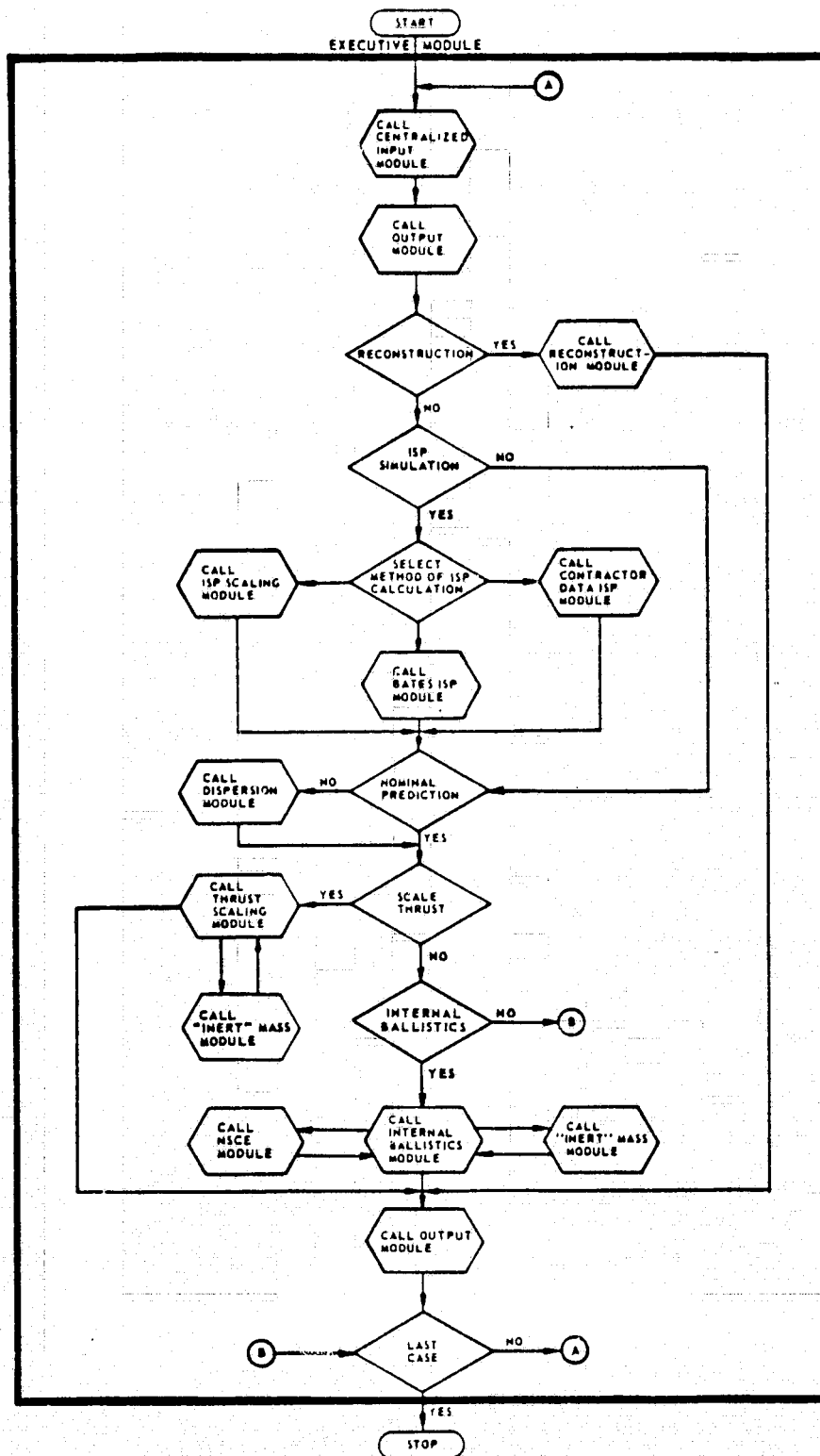


FIGURE 1-1 MODULARIZED SRB-II PROGRAM FLOWCHART

SECTION 2

2.0 EXECUTIVE MODULE

The Executive Module controls access to the computation modules so that a unified analysis of solid rocket booster performance can be performed. Since the Executive Module is a control routine, no computations are performed in the module other than accounting for the number of cases executed. The control options which the user may exercise are given in Volume II of this report. A list of subroutines and modules called by the Executive Module is given in Table 2-I, along with the paragraph in which detailed descriptions of each are found.

TABLE 2-I: SUBROUTINES AND MODULES CALLED BY THE EXECUTIVE MODULE

<u>SUBROUTINE OR MODULE NAME</u>	<u>PARAGRAPH NO.</u>
BATES	4.0
CDSI	6.0
DISP	11.0
FSCAL	8.0
IBM	7.0
INIT1	2.2
INPUT	3.0
OUTPUT	12.0
OUTTAP (Entry to OUTPUT)	12.0
PRESET	2.3
RECON	10.0
SISCAL	5.0

2.1 EXECUTIVE MODULE ENGINEERING DESCRIPTION

The case counter, NCASE, is initialized to 0 and Subroutine PRESET is called to initialize program option indicators to their fail-safe values. PRESET is called only once during the program execution to set the fail-safe indicator values (Note that changes in these indicator values in any early case will be retained for subsequent cases unless updated).

The Centralized Input Module, INPUT, is called in Statement 10 to read all input data from cards for the present execution case and transfer these data to disk. An initialization routine, INIT1 is then called to set initial values for parameters which are required to be reset in each case execution.

The internal program indicator NPRINT is set to 1 to require the Output Module to list all initial values for the case to be executed. This initial print is generated by the Output Module in the call to OUTPUT.

2.1 (Continued)

The indicator NRECON is set to 0 for prediction and to 1 for reconstruction. If NRECON is greater than zero, a reconstruction analysis has been requested and the program branches to Statement 100. At Statement 100 the print indicator NPRINT is set to 8 to indicate a reconstruction output format and the Reconstruction Module (RECON) is called to perform the analysis.

The computed GO TO of Statement 20 allows the user to choose one of the four possible routes through the specific impulse and characteristic exhaust velocity scaling prediction portion of the program. Choice of these routes is controlled by the external input indicator NSI. For NSI=1 the program branches to Statement 30 where the internal print indicator NPRINT is set to 2 such that specific Impulse Scaling Module output format is generated by the Output Module. The Specific Impulse Scaling Module (SISCAL) is then called to calculate the End Item motor delivered vacuum specific impulse and the characteristic exhaust velocity curve fit coefficients.

For NSI=2 the program branches to Statement 40 and the print indicator NPRINT is set to 3 such that Contractor Data Specific Impulse Scaling Module (CDSI) output format is generated by the Output Module. CDSI is then called to calculate the characteristic exhaust velocity curve fit coefficients and the End Item motor delivered vacuum specific impulse.

The program branches to Statement 50 for NSI=3 and the internal print indicator NPRINT is set to 4 for BATES Specific Impulse Module (BATES) output format. BATES is then called to calculate the characteristic velocity curve fit coefficients and the End Item motor delivered vacuum specific impulse. The variable ERR is set by the DATLOC subroutine called by the BATES Module. If ERR is .TRUE., the required data for a BATES prediction was not included in the input data deck. For the condition ERR=.TRUE., the case is immediately terminated and the next case is initiated.

In the case of NSI set to 4, the program branches directly to Statement 60 and none of the specific impulse modules are executed. In any of these events (NSI=1,2,3,4), the test at Statement 60 for a dispersion prediction (NDISP.EQ.0) is executed. A nominal prediction is generated in the case of NDISP.EQ.0 and a test is performed at Statement 80 to determine whether the Internal Ballistics Module or the Thrust Scaling Module (FSCAL) is to be executed to perform a prediction. If the external program input indicator NF is equal to 0, FSCAL is executed. For NF greater than 0, the Internal Ballistics Module (IBM) is executed. If NDISP.GT.0, NPRINT is set to 5 such that the dispersion print format is generated by the Output Module. The Dispersion Module (DISP) is then called to perturb the IBM input parameters for a dispersion analysis and the program proceeds as if a nominal prediction is to be made.

2.1 (Continued)

Statement 110 to Statement 120 increments the executed case count, puts an End of File on the plot tape if this tape is requested, calls OUTTAP to close out the data tape if this tape is generated, and ends the run by branching to Statement 120 when the specified number of cases have been executed. Statement 120 writes the successful completion message formatted in Statement 130.

2.2 SUBROUTINE INIT1 ENGINEERING DESCRIPTION

Subroutine INIT1 is used to initialize several variables used in the "INERT" Mass Module prior to initiating each data case.

The first part of the subroutine sizes and locates the real and common variables used in the subroutine. The subroutine then sets the values of the consumed mass of inerts (MFIHIT), inert mass flowrate (MIFZ), inert thrust contribution (FIZ), and delivered total impulse (AJTIN) to zero. The initial value of the inert mass remaining (MIZ) is set equal to the initial inert mass on board (MITOT)

2.3 SUBROUTINE PRESET ENGINEERING DESCRIPTION

Subroutine PRESET is used to initialize several variables to their fail-safe values for a run. PRESET is called only once per run. Values which are changed from their fail-safe values in any of the executed cases are not reset to their fail-safe values in subsequent cases.

The first part of the subroutine sizes and locates the real and common variables used in the subroutine. The subroutine then sets each of the indicator variables to the values shown in the dashed line boxes in Figures 2-1 and 2-2. Each of these indicator variables will be assigned the value given in the Figures unless a value is input through the NAMELIST method as defined in paragraph 3.1.

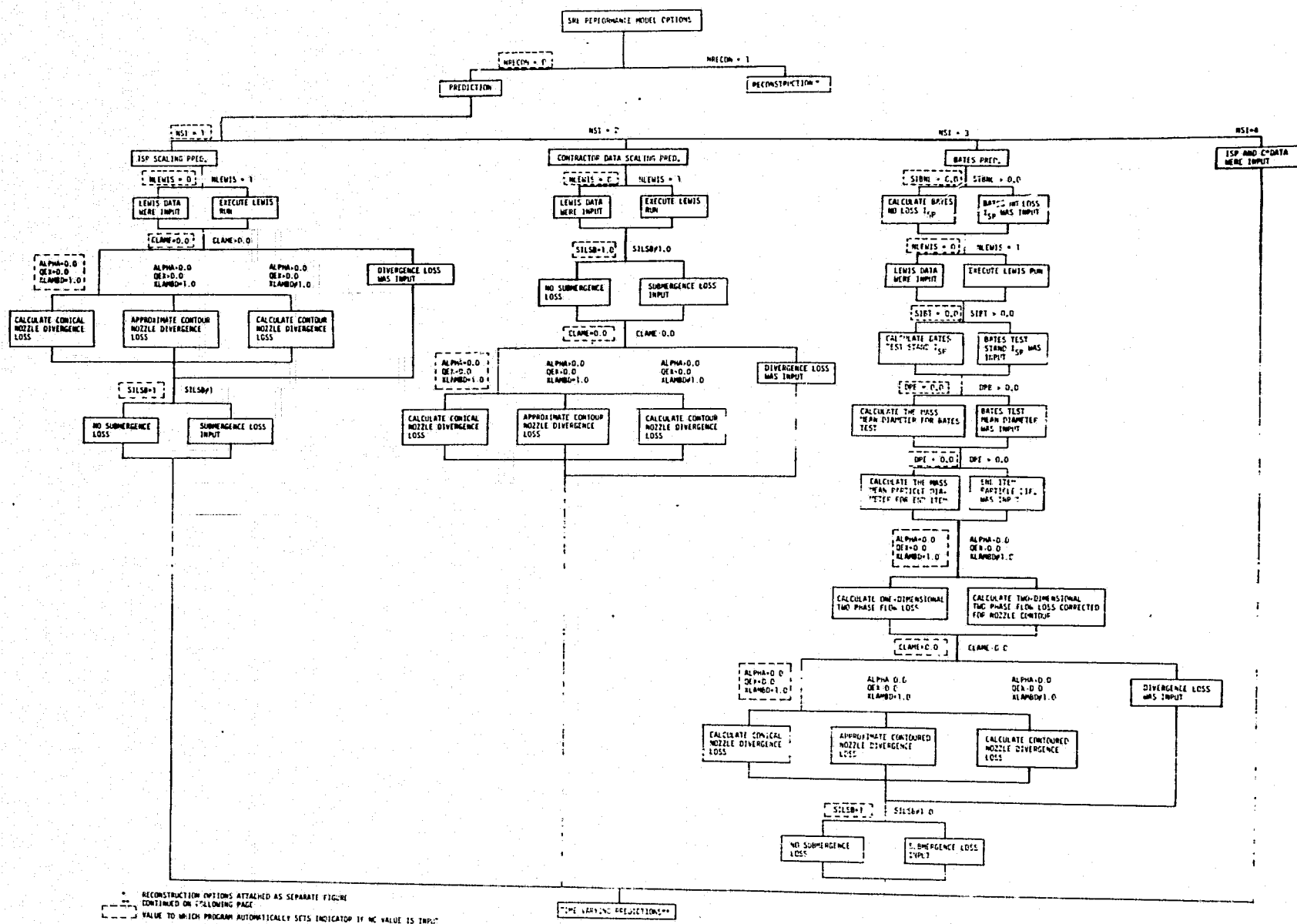
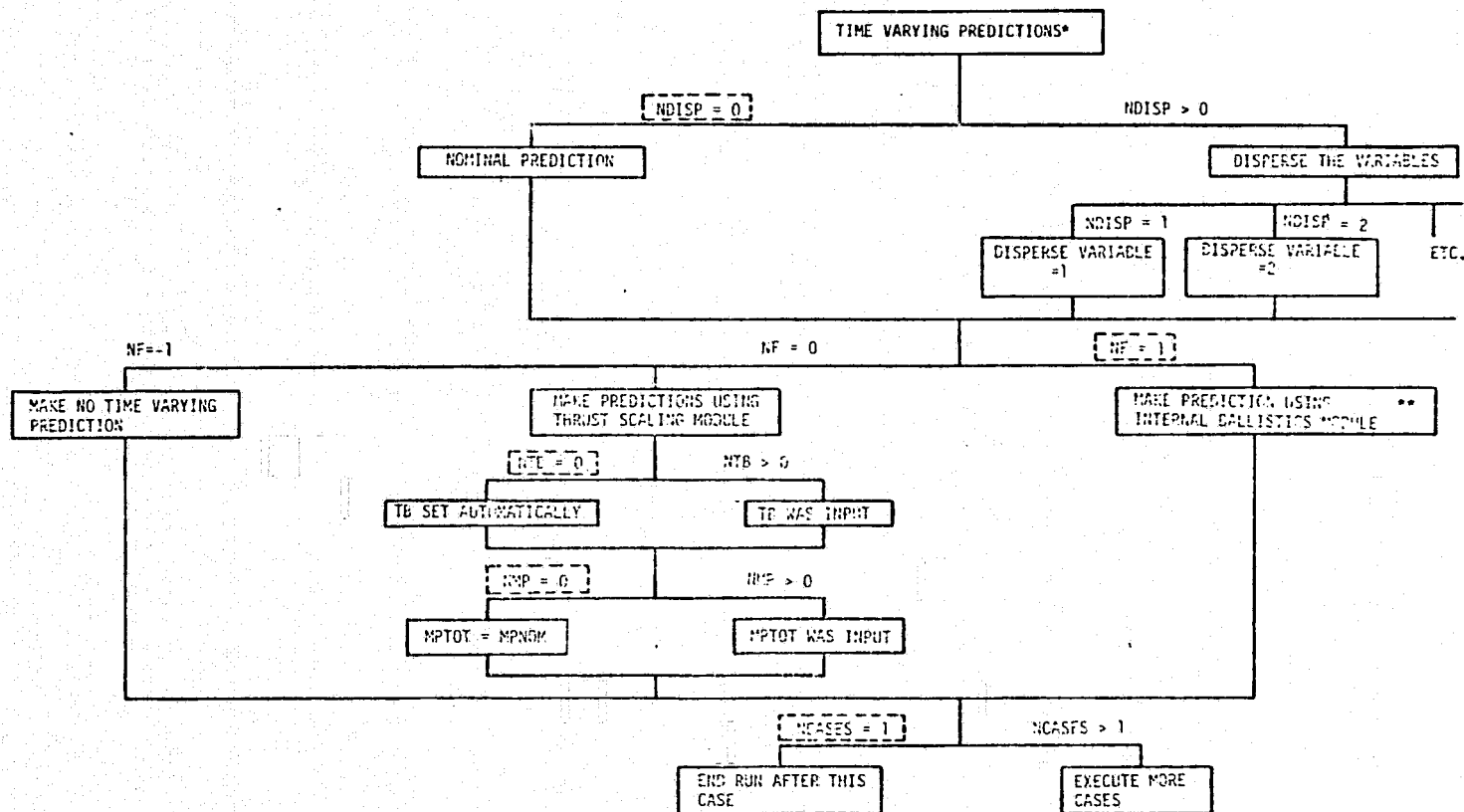


FIGURE 2-1 OPTIONS TREE FOR THE PREDICTION MODE OF SRB-II



* CONTINUED FROM PREVIOUS PAGE

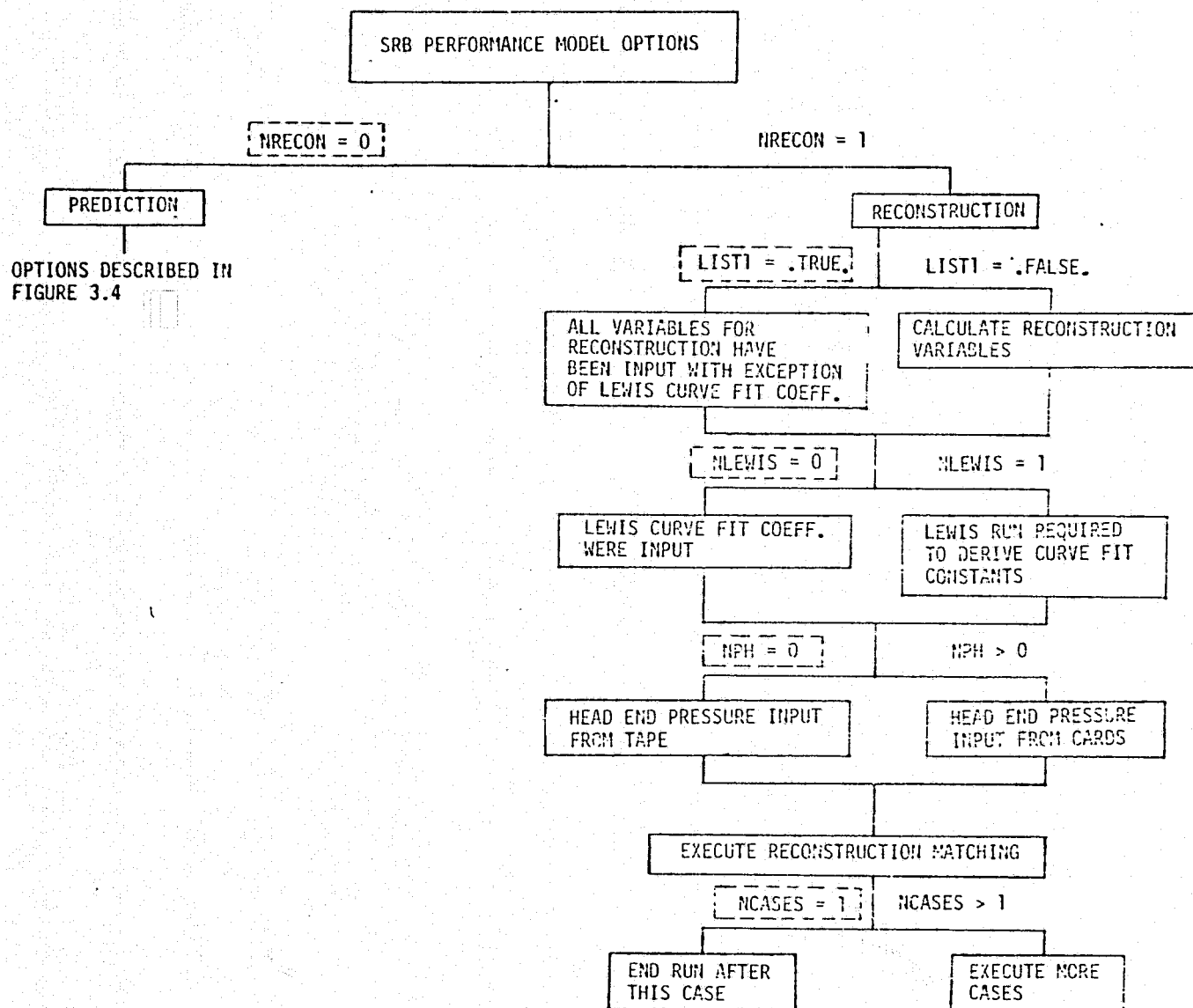
** OPTIONS LIST FOR INTERNAL BALLISTICS ATTACHED AS SEPARATE FIGURE (PARAGRAPH 3.5)

[] VALUE TO WHICH PROGRAM AUTOMATICALLY SETS INDICATOR IF NO VALUE IS INPUT

FIGURE 2-1 OPTIONS TREE FOR THE PREDICTION MODE OF SRB-II (CONTINUED)

D256-10020-1

ORIGINAL PAGE IS
OF POOR QUALITY



 Value to Which Program Automatically Sets Indicator if no Value is Input.

FIGURE 2-2 OPTIONS TREE FOR THE RECONSTRUCTION MODE OF SRB-7

SECTION 3

3.0 CENTRALIZED INPUT MODULE

SRB-II has a built in set of input data which includes all the data needed for program execution. The various input parameters for calculating solid rocket booster performance are contained in Subroutine BLOCK DATA and the program option indicators are contained in Subroutine PRESET. The Centralized Input Module uses Subroutine INPUT to override the data in Subroutine BLOCK DATA and PRESET with the NAMELIST method. Use of the BLOCK DATA and PRESET subroutines in conjunction with the NAMELIST method minimizes the number of input data cards and avoids recompiling SRB-II when a data change is necessary.

SRB-II options that require Subroutine LEWIS utilize the Centralized Input Module to read a special set of input data required for Subroutine LEWIS. Propellant chemistry data is read from a LEWIS data package of set format cards and the propellant thermodynamic data is read from a Lewis-Thermo-Data Tape. This tape is read with Subroutine SEARCH which is a part of Subroutine LEWIS.

SRB-II flight or static test reconstruction option requires a data tape as input. The Centralized Input Module reads this tape with Subroutine GETDAT.

The following paragraphs will describe Subroutine INPUT and GETDAT. Subroutine SEARCH is described in the available documentation for the Lewis Thermochemical Computer Program, Reference 1.

3.1 SUBROUTINE INPUT ENGINEERING DESCRIPTION

The statements after the subroutine name and before Statement 5 are variable location assignments, size, and type (floating-point or fixed) of variables, and the initialization of some variables. This section also contains the list of variables available in the INPUT1 NAMELIST.

Statements 5 through 35 transfer input data cards for a case to disk and print each card. The variable NOUT is the internal unit number (10) for the disk. The array A is a temporary working area to store the input card characters (80) until transferring them to disk and print.

Statements between 35 and 30 contain the logic and write statements to output only the Internal Ballistics Module input data package on disk (internal unit NIBOUT=20) for the first case only. This IBM data package is written on disk for recall during subsequent cases. Statement 30 writes an end of file mark on the disk which contains all input data for a case.

3.1 (Continued)

Statements 40 to 50 rewind the NOUT disk file, test if this is first case and writes an end of file on the IBM data package disk file (NIBOUT), and sets indicator NIBFLG if the IBM data package has been written on disk.

Statements 50 to 70 searches for the CONTROL DATA PACKAGE on disk. When the proper identification card is found, the INPUT1 namelist is read. If the proper identification card is not found, the program directs control to statement 110 and writes an error message, rewinds the disk unit NOUT, and returns control to Statement 5 to read in the next case, if available.

Statements 70 to 80 searches the input data file on disk for the types of DATA PACKAGES available. As each package is found an indicator is set (e.g. NL=1, NIDN=1, and/or NIBM=1).

Statement 80 and the two statements following are tests to determine if there is sufficient data available to run the type of case requested by the indicators set in the CONTROL DATA Package/INPUT1 namelist. If there is not sufficient data the program will direct control to Statement 130 and write an error message, rewind the disk unit NOUT, and return control to Statement 5 to read in the next case, if available.

Statements 90 through 100 generate a header (SPACE SHUTTLE SOLID ROCKET BOOSTER PERFORMANCE EVALUATION MODEL) and a title for the particular case. This information is written on a separate page, centered, and in an enclosed box surrounded by asterisks.

The two statements after Statement 100 will rewind the disk unit NOUT and then returns control to the calling module which is the Executive Module (MAIN).

The Statements 110 through 140 are the error messages explained above.

The Statements 150 through the STOP statement will print a message that all input data cases have been completed and the program will end execution. Statement 150 is reached from Statement 10 when an end of file mark is recognized in the input data card stream.

3.2 SUBROUTINE GETDAT ENGINEERING DESCRIPTION

The two statements after the subroutine name are variable location assignments, size, and type (floating-point or fixed) of variables.

The next two statements are tests on program reconstruction time. If time is zero the subroutine will execute statements 10 through the return of statement 25. These statements rewind the reconstruction tape

3.2 (Continued)

and position it at time equal zero on the tape. If the reconstruction time is less than zero an error message is printed by statements 90 and 100 and the program stops.

When reconstruction time is greater than zero, program control is directed to statement 50. Statement 50 reads ten parameters off the reconstruction tape. The next statement checks the first parameter off the tape, which is time, to see if it is equal to or greater than program time. If so, it executes statements 30 and 40, which is a linear interpolation algorithm. The subroutine then executes a return. If the time off tape is less than program time the subroutine executes the DO 60 loop storing the parameters off tape into array SAVE and then reads the next record off tape. If an end of file is found on the reconstruction tape an error message is printed by statements 70 and 80 and the program stops.

SECTION 4

4.0 BATES ISP MODULE (BATES)

The BATES Isp Module computes the End Item motor vacuum specific impulse (SIDE) and characteristic exhaust velocity (CSTAR) using the Air Force devised Ballistics Test Evaluation Scaling (BATES) method described in Reference 2. The BATES system test procedure employs the use of a carefully designed solid propellant test motor. Propellant samples are prepared by the solid rocket motor contractor in separate cartridges which are subsequently loaded into a test motor. The motor is fired in a calorimeter capable of precise heat loss measurement. The BATES philosophy involves developing the End Item motor specific impulse by correcting the specific impulse derived from the BATES test motor.

BATES Isp Module is selected as the method for calculating End Item motor vacuum specific impulse and characteristic exhaust velocity by setting NSI=3. The module uses the subroutines given in Table 4-I.

TABLE 4-I: BATES ISP MODULE SUBROUTINES

SUBROUTINE NAME	PARAGRAPH NO.
ITIDNZ	4.2
DATLOC(IDNOZ)	4.2
LEWIS & LEWIT	4.3
IDNOZL	4.4
LESSQ	4.5
OUTPUT	12.0

Each of these subroutines except OUTPUT will be described in this Section. Subroutine OUTPUT is described in paragraph 12.0 and will not be repeated here.

4.1 BATES ISP MODULE ENGINEERING DESCRIPTION

The first part of the module, down to Statement 10, sizes and locates the Real and Common Variables used in the module. Subroutine ITIDNZ is called at Statement 10 to initialize the propellant data that will be required later for Subroutine IDNOZL to calculate the two phase flow performance of the BATES and End Item motors.

The test for SIBNL.G.T.O.O is used to calculate the End Item motor specific impulse correction and skip the BATES test motor no loss specific impulse correction if the BATES correction is available for input. The following paragraphs will describe the options that are available for developing End Item motor delivered specific impulse.

4.1.1 BATES Test Motor Performance

The BATES test motor "ideal" or no-loss delivered specific impulse at standard conditions (1000 psia nozzle throat total pressure and 14.696 psia ambient) is obtained by correcting the BATES test actual specific impulse to standard conditions and adding three major specific impulse

4.1.1 (Continued)

correction factors; divergence, heat flow and two phase flow losses. This correction is made with the following equation before Statement 70.

$$SIBNL = SIBT + SILDVB + SILQB + SIL2PB$$

The variable SIBT is the BATES test motor specific impulse corrected to standard conditions, SILDVB is the BATES test motor divergence loss, SILQB is BATES test motor heat flow loss, and SIL2PB is the BATES motor two phase flow loss. The method used to calculate each of these variables will be given in the following paragraphs.

At Statement 40 the BATES test motor actual specific impulse is corrected to standard conditions by the ratio of the theoretical specific impulse at standard conditions to actual test conditions, SIAS/SIBAT or after Statement 30 by using an empirical efficiency factor (ETABT) which must be input. The values of theoretical specific impulse used in this correction may be developed with the curve fit algorithm or with the Subroutine LEWIS. The curve fit algorithm was developed from theoretical nozzle performance equations as follows:

THEORETICAL THRUST

$$F \equiv \dot{m} V_e + A_e (P_e - P_a)$$

THEORETICAL CHARACTERISTIC EXHAUST VELOCITY

$$C^* \equiv \frac{P_c A_t}{\dot{m}}$$

THEORETICAL SPECIFIC IMPULSE

$$\frac{F}{\dot{m}} = V_e + \frac{A_e}{\dot{m}} (P_e - P_a)$$

or

$$\frac{F}{\dot{m}} = V_e + \frac{A_e}{A_t P_c} C^* (P_e - P_a)$$

rearranging

$$\frac{F}{\dot{m}} = V_e + \frac{A_e}{A_t} \frac{P_e}{P_c} C^* - \frac{A_e}{A_t} \frac{P_a}{P_c} C^*$$

4.1.1 (Continued)

simplify

$$I_{sp} = I_{vac} - \frac{A_e}{A_t} \frac{p_a}{p_c} c^*$$

Using this equation form, the curve fit algorithm for theoretical specific impulse at standard conditions (SIAS) and actual conditions (SIBAT) were developed. The equation used to calculate SIAS is as follows.

$$SIAS = AK(1) + AK(2) * MC(3) + AK(3) * MC(3) ** 2 - MC(6) * MC(3) * CK(2) / (MC(4) * CK(1))$$

In this equation the curve fit constants for theoretical vacuum specific impulse AK(1), AK(2), and AK(3) are input, MC(3) is the optimum expansion area ratio for 1000 psia to 14.696 psia, MC(6) is the propellant theoretical characteristic exhaust velocity at 1000 psia chamber pressure, MC(4) is the BATES test motor chamber pressure, CK(2) is the standard ambient pressure 14.696 psia, and CK(1) is the gravitational constant, g(32.174 FT/SEC). The equation used to calculate SIBAT is as follows:

$$SIBAT = AK(1) + AK(2) * ARETB + AK(3) * ARETB ** 2 - ARETB * MC(2) * PAMBT / (PCBT * CK(1))$$

The constants in this equation are the same as used in the equation for SIAS, ARETB is the BATES test motor expansion ratio, PAMBT is the ambient pressure for BATES test, and PCBT is the BATES test motor nozzle throat total pressure.

Subroutine LEWIS calculates both SIAS and SIBAT when NLEWIS is set to 1.0. Prior to assigning values to nozzle throat total pressure, nozzle pressure ratio, and area ratio required for input to Subroutine LEWIS it is necessary to use the initialization routine Subroutine LEWIT. Subroutine LEWIT assigns zero to all previous values used for these variables. Beginning at Statement 20 new values for BATES test motor standard chamber pressure, a random set of nozzle throat total pressures, standard expansion ratio, BATES test motor nozzle area ratio, and End Item motor nozzle area ratio are set for input to Subroutine LEWIS. The End Item motor data are used here to calculate End Item motor specific impulse (SIAE). This eliminates the need to call Subroutine LEWIS a second time later in the Module. This call to Subroutine LEWIS also calculates theoretical characteristic exhaust velocity for several nozzle throat total pressures and the theoretical specific impulse at the nozzle throat which are to be used later in the calculation for End Item motor CSTAR as a function of nozzle throat total pressure.

The BATES test motor specific impulse divergence loss correction (SILDVB) is developed by correcting the theoretical specific impulse

4.1.1 (Continued)

at standard conditions (SIAS) with a constant, MC(5), that is a function of the nozzle divergence half angle using the following equation at Statement 50.

$$\text{SILDVB} = \text{SIAS} * \text{MC}(5)$$

The BATES test motor specific impulse heat flow loss correction (SILQB) is calculated with the following standard heat loss equation after Statement 50.

$$\text{SILQB} = \text{SII} - \text{SILDVB} - \text{SIBT}$$

$$\text{SII} = ((\text{SIBT} + \text{SILDVB})^{**2} + 2.0 * \text{CK}(3) * \text{QBT} / (\text{CK}(1) * \text{WPBT}))^{**0.5}$$

In this equation (SIBT) is the BATES test motor specific impulse at standard conditions and (SILDVB) is the specific impulse nozzle divergence loss correction that was calculated above. The mechanical equivalent of heat (CK(3)) is 778 FT-LB/BTU and the gravitational constant (CK(1)) is 32.174 FT/SEC. The heat loss measured with the BATES test calorimeter is defined as (QBT). The weight of propellant used in the BATES test motor is (WPBT).

The BATES test motor two phase flow loss correction is calculated with the following equation above Statement 70.

$$\text{SIL2PB} = \text{SIAS} * (1 - \text{RTOI}(1))$$

In this equation the BATES test motor theoretical specific impulse at standard conditions (SIAS) is that calculated above. The ratio of the lag to no lag specific impulse for an optimum expansion (RTOI(1)) is calculated in Subroutine IDNOZL. Subroutine IDNOZL calculates various parameters describing two phase flow through a nozzle using the input BATES motor geometry and the propellant data read in through Subroutine ITIDNZ. The BATES motor combustion product solid particle mean diameter (DPB) may be input or calculated as a function of the nozzle diameter (DTB). The relationship of DPB and DTB is a fifth order polynomial which was derived in Reference 3.

4.1.2 End Item Motor Performance

If the BATES test motor no loss specific impulse (SIBNL) is available for input, the program begins calculation of the End Item motor theoretical specific impulse at vacuum conditions (SIAE) and standard conditions (SIAS) at Statement 70. The two options available for calculating SIAE and SIAS are curve fit algorithms and with Subroutine LEWIS. The methods available for calculating (SIAS) were described in Paragraph 4.1.1. The curve fit algorithm for (SIAE) expansion ratio is as follows.

4.1.1.2 (Continued)

$$SIAE = AK(1) \cdot ARETE + AK(3) \cdot ARETE^2$$

In this equation the constants $AK(1)$, $AK(2)$, and $AK(3)$ are input and $ARETE$ is the End Item motor nozzle area ratio. The option to use Subroutine LEWIS to calculate $ARETE$ is also available and is identical to the method described in Paragraph 4.1.1.

Beginning at Statement 20 the BATES test motor "ideal" or no loss delivered specific impulse ($SIENL$) will be used as a basis for developing the End Item motor delivered specific impulse ($SIDE$) and characteristic exhaust velocity ($CSTAR$). This delivered vacuum specific impulse at 1000 psia nozzle throat total pressure is calculated at Statement 120 with the following equation.

$$SIDE = (SIENL - SIL2PE - SILOE - SILDVE) \cdot SILSB$$

In this equation $SIENL$ is the End Item motor no loss vacuum specific impulse at 1000 psia nozzle throat total pressure, $SIL2PE$ is the End Item motor two phase flow loss, $SILOE$ is the End Item motor heat flow loss, $SILDVE$ is the End Item motor divergent loss, and $SILSB$ is the End Item motor nozzle submergence loss factor. The methods used to calculate each of these variables will be given in the following paragraphs.

The End Item motor no loss vacuum specific impulse $SIENL$ is calculated with the following equation.

$$SIENL = SIBNL \cdot SIAE / SIAS$$

In this equation the BATES test motor no loss specific impulse at standard condition ($SIBNL$) is corrected to End Item Motor area ratio and vacuum conditions by the ratio of theoretical specific impulse at vacuum conditions to standard conditions, $SIAE/SIAS$. The calculation procedure for $SIAE$ and $SIAS$ are identical to those described above.

The End Item motor two phase flow loss ($SIL2PE$) is calculated with the following equation if the motor has a conical nozzle.

$$SIL2PE = SIAE \cdot (1 - RT012)$$

The End Item motor theoretical specific impulse at vacuum conditions ($SIAE$) was calculated above. The ratio of the lag to no lag specific impulse for an optimum expansion ($RT012$) is calculated with Subroutine IDNOZL. Solid particle mean diameter, propellant characteristics, and nozzle geometry input data are required for the End Item motor. The solid particle mean diameter (DPE) for the End Item motor may be input or calculated as a function of nozzle throat diameter (DTE) with the fifth order polynomial described in Paragraph 4.1.1.

4.1.2 (Continued)

When the End Item motor has a contoured nozzle the two phase flow loss (SIL2PE) can be calculated with the following equation at Statement 105.

$$SIL2PE = SIAE * (1. - (RTOI2 * ((0.5 - 0.5 * \cos(\alpha)) / (0.5 - 0.5 * \cos((\alpha - QEX) / 2.))) * XLAMBD))$$

The End Item motor theoretical specific impulse at vacuum conditions (SIAE) and the ratio of the lag to no lag specific impulse for an optimum expansion (RTOI2) are those calculated above. ALPHA is the equivalent conical divergence angle of the contoured nozzle and QEX is the slope of the nozzle contour at the exit plane. The ratio of the theoretical specific impulse of the contoured nozzle to the theoretical specific impulse of an equivalent conical nozzle (XLAMBD) is derived with the Axisymmetric Two-Phase Perfect Gas Performance Program (ATPAP) described in Reference 4. Since ATPAP is a large program and will be required only once during a series of predictions it will be executed separately to develop XLAMBD for input data for SRB-II. ATPAP is currently operational at MSFC on the Univac 1108 and at Boeing on the IBM 370.

The End Item motor specific impulse heat flow loss correction (SILQE) is calculated with the following equations.

$$SILQE = SIENL - SIL2PE - SI2$$

$$SI2 = ((SIENL - SIL2PE) ** 2 - QAIE * WINE * 2 * CK(3) / (WPE * CK(1))) ** 0.5$$

SIENL and SIL2PE are as defined and calculated above. The two constants CK(1) and CK(3) are the gravitational constant (32.174 FT/SEC) and mechanical equivalent of heat (778 FT-LB/BTU). WINE is the mass of expended inert liner material and QAIE is the heat of ablation of this material in BTU/LB. The End Item motor propellant weight is WPE.

The End Item motor specific impulse divergence loss correction (SILDVE) is calculated with the following equation if the motor has a conical nozzle.

$$SILDVE = SIAE * (1.0 - CLAME)$$

This equation uses the End Item motor nozzle divergence half angle AHALFE to calculate the correction applied to the End Item motor theoretical specific impulse at vacuum conditions (SIAE) that was calculated above.

The End Item motor divergence loss correction (SILDVE) for a contoured nozzle can be calculated with one of two options. The approximate correction method uses the following equation at Statement 115.

4.1.2 (Continued)

$$SILDVE = SIAE * (0.5 - 0.5 * \cos((\text{ALPHA} + \text{QEX})/2.0))$$

In this equation SIAE, ALPHA and QEX are those defined or calculated above. The method of characteristic correction uses the following equation at Statement 117.

$$SILDVE = SIAE * (0.5 - 0.5 * \cos(\text{ALPHA})) * \text{XLAMBD}$$

In this equation SIAE, ALPHA and XLAMBD are those defined and calculated above.

The End Item motor characteristic exhaust velocity (CSTAR) is calculated with the following equation below Statement 120.

$$\text{CSTAR} = \text{OK}(1) * \text{CSTAR1} * ((\text{SIENL} - (1.0 - \text{RTOI3}) * \text{SI3} - \text{SILOE}) * \text{SILSB}) / \text{SIAE}$$

In this equation the theoretical characteristic exhaust velocity at 1000 psia nozzle throat total pressure (CSTAR1) is adjusted to show the losses prior to the nozzle throat. These losses include the heat loss and the portion of the two phase flow loss that occurs up to the nozzle throat. SIENL, SILQUE, SILSB, and SIAE are the values defined or calculated above for the End Item motor. The two phase flow specific impulse loss up to the throat (RTOI3) was calculated in Subroutine IDNOZL when it was used above to calculate RTOI2. The theoretical specific impulse at the nozzle throat (SI3) was calculated in Subroutine LEWIS when it was used above to calculate SIAE and SIAS.

The use of CSTAR in other modules requires CSTAR as a function of nozzle throat total pressure. Two options are provided for developing these data. The second order polynomial given in Statement 130 is used if the option NLEWIS=0 was selected earlier in the Module. In this case, the curve fit coefficients (CSCOE) must be input. If Subroutine LEWIS has been used to calculate CSTAR1 it also calculates several additional theoretical characteristic exhaust velocities for a range of nozzle throat total pressures. Beginning at Statement 140, a routine is used to calculate a second order curve fit for these data to establish a theoretical characteristic exhaust velocity at the zero pressure intercept. The curve fit is done by Subroutine LESSQ. The calculated zero pressure intercept is shifted so that the curve fit will pass through the End Item motor characteristic exhaust velocity CSTAR. The revised intercept and the curve fit constants are set up in an equation for CSTAR as a function of nozzle throat total pressure in the following equation for use in other modules.

$$\text{CSCOE}(1) = \text{CSTAR} - \text{CSCOE}(2) * \text{MC}(4) - \text{CSCOE}(3) * \text{MC}(4) ** 2$$

Data calculated in the BATES Isp Module are transferred to the Output Module where they are printed in a specified format in both English and Metric Units.

4.2 SUBROUTINE ITIDNZ AND DATLOC (IDNOZ) ENGINEERING DESCRIPTION

The initial part of Subroutine ITIDNZ sizes and locates the variables that are required for input to Subroutine IDNOZL. It uses Subroutine DATLOC(IDNOZ) to search the data disk to locate the appropriate input data package that was read on disk with the Input Module. If the data package is missing the Subroutine automatically reads the next case. The items in the input data package are given in Table 3-VIII of Volume II.

4.3 SUBROUTINE LEWIS AND LEWIT ENGINEERING DESCRIPTION

The programming mechanism of calling Subroutine LEWIS requires that Subroutine LEWIT be called to set the initial values of several variables in Subroutine LEWIS. The following variables are set in Subroutine LEWIT.

```
PCP(I) = 0.0
SUBAR(I) = 0.0
SUPAR(I) = 0.0
P(I) = 0.0
IRKT02=1
```

After the values are initialized, Subroutine LEWIS is used to calculate theoretical specific impulse and characteristic exhaust velocity for a given propellant formulation at specified chamber pressures, pressure ratios and nozzle area ratios. Subroutine LEWIS is an adaptation of the CEC71 program developed by NASA Lewis Research Center, Reference 1. CEC71 has been reduced to work only rocket problems, then further reduced to decrease core requirements. The following options and features were removed from CEC71 to develop Subroutine LEWIS.

- a) The number of species (combustion products) which can be considered were reduced from 150 to 115. This eliminates consideration of some of the more complicated propellant combinations. These combinations may be considered by first running the full CEC71 program externally and then submitting results in the LEWIS DATA package.
- b) All NAMELIST input data requirements were removed. This restricts the units and methods by which the data may be transmitted to the LEWIS subroutine.
- c) The option for intermediate output was removed. The capability for debug print in cases of bomb-out and non-convergence no longer exists.
- d) All normal program output other than the specialized rocket parameters printed from RKTOUT were removed. The output from RKTOUT is optional and may be obtained by setting an indicator in the input data.

4.3 (Continued)

- e) The option of entering thermodynamic data from cards was removed. This limits the user to the standard thermodynamic data on the NASA LEWIS supplied data (LEWIS Thermo Data Tape).

Other than the modification described above, the detailed engineering description of Subroutine LEWIS is given in Reference 1.

4.4 SUBROUTINE IDNOZL ENGINEERING DESCRIPTION

Subroutine IDNOZL is an adaptation of the IDNOZL Program developed by Bruce Kubert at Aerospace Corporation in 1964, Reference 5. The subroutine utilizes the basic computer program. Modifications were made only to the input-output schemes where necessary to use the program as a subroutine. Input data for IDNOZL, set up as described above in Subroutines ITIDNZ and DATLOC (IDNOZ), includes motor geometry and propellant data. Subroutine IDNOZL calculates the parameters describing two phase flow through a nozzle. It assumes that the flow is one-dimensional and that the chemical composition of the gas-particle mixture remains fixed throughout the nozzle. The subroutine calculates the lag or no lag specific impulse, which are used in the two phase flow loss calculations. The lag specific impulse is calculated assuming that the gas and particles have a different velocity and temperature throughout the nozzle. The no lag specific impulse assumes that both the gas and particles are at the same velocity and temperature throughout the nozzle. Two phase flow specific impulse losses may be calculated both at the nozzle throat and at the nozzle exit. The detailed Engineering descriptions of the basic equations used in the Subroutine IDNOZL are given in Reference 5.

4.5 SUBROUTINE LESSQ ENGINEERING DESCRIPTION

This algorithm develops a second degree polynomial curve fit for at least three coordinate points using the principle of least squares. The form of the second degree polynomial fit is given in the following equation:

$$y = a_1 + a_2x + a_3x^2$$

The coefficients a_1 , a_2 and a_3 are developed by solving the following equations.

$$a_3 = \frac{(EXZY - NEXY)(EXEX^2 - NEX^3) - (EX^2EY - NEX^2Y)[(EX)^2 - NEX^2]}{(EXEX^2 - NEX^3)^2 - [(EX^2)^2 - NEX^4][(EX)^2 - NEX^2]}$$

4.5 (Continued)

$$a_2 = \frac{(\sum x \sum y - n \sum xy) - a_3 (\sum x \sum x^2 - n \sum x^3)}{[(\sum x)^2 - n \sum x^2]}$$

$$a_1 = \frac{\sum y - \sum x a_2 - \sum x^2 a_3}{n}$$

The values of the coefficients are returned to the Module that has called Subroutine LESSQ.

4.6 BATES ISP MODULE LIMITATIONS

The BATES Isp Module should be used only when a complete set of data are available from a BATES motor test. The two phase flow losses that are calculated with Subroutine IDNOZL for both the BATES and End Item motors are restricted to specific chamber to throat radius ratios. Chamber to throat radius ratio for the BATES motor is fixed at 4.0 and for the End Item motor it is fixed at 3.65352.

SECTION 5

5.0 ISP SCALING MODULE (SISCAL)

The Isp Scaling Module calculates the End Item motor delivered vacuum specific impulse and characteristic exhaust velocity. End Item motor delivered vacuum specific impulse is developed by correcting the theoretical specific impulse for divergence losses, submergence losses, combustion inefficiency, and motor inefficiency as a function of mass flow rate. End Item motor characteristic exhaust velocity is developed by correcting the theoretical characteristic exhaust velocity for combustion inefficiency and submergence losses. This module is selected by setting NSI=1. It should be used for preliminary design iterations since it provides a good estimate of these parameters and requires a minimum of input data.

The module uses the subroutines given in Table 5-1.

TABLE 5-1: ISP SCALING MODULE SUBROUTINES

<u>Subroutine Name</u>	<u>Paragraph Number</u>
LEWIT	4.3
LEWIS	4.4
LESSQ	4.6
OUTPUT	12.0

The engineering description of each of these subroutines is given in other paragraphs and will not be repeated.

5.1 ISP SCALING MODULE ENGINEERING DESCRIPTION

The first portion of the module, down to Statement 20, sizes and locates the Real and Common variables used in the module. Beginning at Statement 20 the module calculates End Item motor vacuum specific impulse at average nozzle throat total pressure by correcting the theoretical vacuum specific impulse (SIAE) for divergence losses, motor inefficiency as a function of mass flow rate, combustion inefficiency and submergence loss. These corrections are made with the following equation at Statement 90.

$$SIDE = CLAME * ETANZ * ETACS * SIAE * SILSB$$

The variable CLAME is the divergence loss factor which is based on the End Item motor nozzle divergence half angle, ETANZ is the motor efficiency factor, ETACS is the motor combustion efficiency factor and SILSB is the submergence loss factor. The method used to develop each of these correction factors will be given in the following paragraphs.

The End Item motor specific impulse divergence loss correction (CLAME) may be either input or calculated. The methods used to calculate CLAME are dependent on whether the End Item motor nozzle is conical or contoured. The following standard equation is used below Statement 70 when the nozzle is conical.

5.1 (Continued)

$$CLAME = 0.5 + 0.5 * \cos(AHALFE)$$

This equation uses the End Item motor nozzle divergence half angle AHALFE to calculate the divergence loss correction factor.

When the End Item motor is contoured two methods are available to the user to calculate CLAME. The approximate method uses the following equation below Statement 70.

$$CLAME = 0.5 + 0.5 * \cos((ALPHA + QEX) / 2.0)$$

In this equation ALPHA and QEX are functions of nozzle geometry that were defined in paragraph 4.1.2. The method of characteristics correction uses the following equation above Statement 90.

$$CLAME = (0.5 + 0.5 * \cos(ALPHA)) * XLAMBD$$

In this equation, ALPHA and XLAMB are those defined in paragraph 4.1.2.

The End Item motor efficiency factor is calculated at Statement 60 with a second order curve fit equation. The curve fit is based on a semi log plot of motor efficiency versus End Item motor mass flowrates and has the following form.

$$ETANZ = (AK(10) + AK(11) * \text{ALOG}(WD1) + AK(12) * (\text{ALOG}(WD1))^2)$$

The curve fit coefficients are AK(10), AK(11) and AK(12). The mass flowrate for the End Item motor (WD1) is calculated at Statement 50 with the following equation that relates characteristic exhaust velocity and the mass flowrate through the nozzle.

$$WD1 = PCAVE * CK(4) * DTE^2 * CK(1) / (4 * CSTAR2 * ETACS)$$

In this equation PCAVE is the average nozzle throat total pressure of the End Item motor. The constants CK(4) and CK(1) are $\pi(3.14169)$ and the gravitational constant $g(32.174 \text{ FT/SEC})$. The End Item motor theoretical characteristic exhaust velocity (CSTAR2) may be input as MC(7) or calculated with Subroutine LEWIS. Subroutine LEWIS calculates CSTAR 2 with the End Item motor average nozzle throat total pressure PCAVE. The motor efficiency factor ETACS is an input parameter that is used in both the equation for WD1 and SIDE.

Two options are available for calculating the End Item motor theoretical vacuum specific impulse SIAE. These options use Subroutine LEWIS and a second order curve fit algorithm of theoretical vacuum specific impulse as a function of nozzle area ratio identical to that shown in paragraph 4.1.2. Prior to assigning values to nozzle throat total pressure and area ratio required for input to Subroutine LEWIS, it is necessary to

5.1 (Continued)

use the initialization routine Subroutine LEWIT. Subroutine LEWIT assigns zero to all previous values used for input to Subroutine LEWIS. The parameters set up for input to Subroutine LEWIS are End Item motor average nozzle throat total pressure, a random set of nozzle throat total pressures, and End Item motor nozzle expansion ratio. This call to Subroutine LEWIS also calculates theoretical characteristic exhaust velocity for the End Item Motor and several nozzle throat total pressures which are required later in the calculation for End Item motor characteristic exhaust velocity as a function of nozzle throat total pressure. The curve fit algorithm for theoretical vacuum specific impulse at End Item motor average nozzle throat total pressure as a function of nozzle expansion ratio is as follows.

$$SIAE = AK(1) + AK(2) * ARETE + AK(3) * ARETE ** 2$$

In this equation the curve fit constants are AK(1), AK(2) and AK(3). ARETE is the End Item nozzle expansion ratio.

The End Item motor submergence loss correction (SILSB) is required input data. The value of this empirical correction should be derived with applicable data from tests specifically conducted to determine the effects of nozzle submergence.

The End Item motor characteristic exhaust velocity CSTAR is calculated with the following equation.

$$CSTAR = CSTAR2 * ETACS * OK(3) * SILSB$$

In this equation the theoretical characteristic exhaust velocity (CSTAR2) is adjusted by a motor efficiency factor (ETACS) and the submergence factor. CSTAR2 was calculated above when Subroutine LEWIS was called and is based on the End Item motor average nozzle throat total pressure (PCAVE) and expansion ratio (ARETE). An empirical factor (OK(3)) is provided in the event it is needed to further adjust CSTAR2 to match an actual motor. The submergence loss correction factor (SILSB) is that defined above.

The use of CSTAR in other modules requires CSTAR as a function of nozzle throat total pressure. Two options are provided for developing these data. The second order polynomial given in Statement 130 is used if Subroutine LEWIS was not used earlier in the Module. In this case, the curve fit coefficients (CSCOE) must be input. If Subroutine LEWIS has been used to calculate CSTAR2 it also calculates several additional theoretical characteristic exhaust velocities for a range of nozzle throat total pressures. Beginning at Statement 140, a routine is used to calculate a second order curve fit for these data to establish a theoretical characteristic exhaust velocity at the zero pressure intercept. This curve fit is performed by Subroutine LESSQ. The calculated zero pressure intercept is shifted so that the curve fit will pass through the End Item

5.1 (Continued)

motor characteristic exhaust velocity (CSTAR). The revised intercept and the curve fit constants are set up in an equation for CSTAR as a function of End Item motor average chamber pressure in the following equation.

$$\text{CSCOE}(1) = \text{CSTAR} - \text{CSCOE}(2) * \text{PCAVE} - \text{CSCOE}(3) * (\text{PCAVE} ** 2)$$

The data calculated in the Isp Scaling Module are transferred to the Output Module where they are printed in a specified format in both English and Metric units.

SECTION 6

6.0 CONTRACTOR DATA ISP MODULE (CDSI)

The Contractor Data Isp Module is used to calculate End Item motor vacuum specific impulse and characteristic exhaust velocity when contractor small motor test data for the exact End Item motor propellant composition are available. This module converts small motor delivered specific impulse to End Item motor vacuum specific impulse by correcting the small motor actual specific impulse to vacuum conditions, then correcting for differences in nozzle divergence losses and differences in motor efficiency. The End Item motor characteristic exhaust velocity is developed by correcting the test motor actual characteristic exhaust velocity for differences in motor efficiency. This module is selected by setting NSI=2.

The Contractor Data Isp Module uses the Subroutines given in Table 6-I.

TABLE 6-I: CONTRACTOR DATA ISP MODULE SUBROUTINES

<u>Subroutine Name</u>	<u>Paragraph Number</u>
LEWIT	4.3
LEWIS	4.4
LESSQ	4.6
OUTPUT	12.0

The engineering description for each of the Subroutines is given in other paragraphs and will not be repeated.

6.1 CONTRACTOR DATA ISP MODULE ENGINEERING DESCRIPTION

The first portion of the module sizes and locates the Real and Common variables used in the module. The remainder of the module is set up to calculate End Item motor vacuum specific impulse and characteristic exhaust velocity. End Item motor vacuum specific impulse is developed by correcting the test motor actual specific impulse to vacuum conditions then correcting for nozzle divergence loss and motor efficiency differences. These corrections are made with the following equation.

$$SIDE = SICON * (SIAE / SIACON) * (CLAME / CLAM1) * OK(2) * (ETANZ / ETANZ1)$$

In this equation SICON is the test motor actual specific impulse, SIACON is the theoretical test motor specific impulse at test conditions, SIAE is the End Item motor theoretical vacuum specific impulse, CLAME is the End Item motor nozzle divergence loss correction, CLAM1 is the test motor nozzle divergence loss correction, ETANZ is the End Item motor efficiency factor, ETANZ1 is the test motor efficiency factor, and OK(2) is an empirical factor for adjusting the motor efficiency ratio. The method used to calculate each of these variables will be given in the following paragraphs.

6.1 (Continued)

The test motor actual specific impulse SICON is assumed to be available for the contractor test motor. It is also assumed that the test motor propellant is identical to the propellant to be used in the End Item motor and nozzle throat total pressures of both motors are essentially the same.

The initial correction to SICON is performed with the ratio of the End Item motor theoretical vacuum specific impulse to the test motor theoretical specific impulse at test conditions, SIAE/SIACON. There are two methods available for developing these variables. These are a curve fit algorithm or Subroutine LEWIS. When the curve fit algorithm is selected, SIAE is calculated as a function of End Item motor nozzle expansion ratio, ARETE, and theoretical vacuum specific impulse at End Item motor average nozzle throat total pressure (PCAVE) with the following equation.

$$SIAE = AK(1) + AK(2) * ARETE + AK(3) * ARETE ** 2$$

The curve fit constants AK(1), AK(2) and AK(3) in this equation are required input data.

The curve fit algorithm for SIACON is developed from the following theoretical nozzle performance equation derived in paragraph 4.1.1.

$$I_{sp} = I_{vac} - \frac{A_e}{A_t} \frac{P_a}{P_c} c^*$$

Using this equation form, the following equation for the test motor theoretical specific impulse SIACON was developed.

$$SIACON = AK(13) + AK(14) * ARET + AK(15) * ARET ** 2 - ARET * MC(23) * PAMT / (PCT * CK(1))$$

In this equation the theoretical vacuum specific impulse curve fit constants AK(13), AK(14), and AK(15) based on test motor nozzle throat pressure (PCT) are input. ARETE is the test motor nozzle expansion ratio, MC(23) is test motor characteristic exhaust velocity at the actual chamber pressure, PAMT is the test condition ambient pressure, PCT is the test motor nozzle throat total pressure, and CK(1) is the gravitational constant, g(32.174 FT/SEC²).

When Subroutine LEWIS is selected to calculate values for the test motor specific impulse correction SIAE/SIACON, it is necessary to use the initialization routine Subroutine LEWIT. Subroutine LEWIT assigns zero to all previous values used for input to Subroutine LEWIS. The parameters set up for input to Subroutine LEWIS are test motor and End Item motor nozzle throat total pressure and nozzle expansion ratio. This call to Subroutine LEWIS also calculates theoretical characteristic exhaust velocity for several nozzle throat total pressures which are required later in the calculation for End Item motor characteristic exhaust

6.1 (Continued)

velocity as a function of nozzle throat total pressure. Since Subroutine LEWIS calculates only test motor theoretical vacuum specific impulse SILVAC(1) it is necessary to use the following equation to calculate test motor theoretical specific impulse at test conditions.

$$SIACON = SILVAC(1) - ARET * MC(23) * PAMT / (PCT * CK(1))$$

The variables used in this equation to correct the theoretical vacuum specific input are identical to those used in the curve fit equation for SIACON.

The nozzle divergence loss difference correction, CLAME/CLAM1 is developed by the following standard equations.

$$CLAME = 0.5 + 0.5 * \cos(AHALFE)$$

and

$$CLAM1 = 0.5 + 0.5 * \cos(AHALF1)$$

These equations use the End Item motor conical nozzle divergence half angle, AHALFE and the test motor conical divergence half angle, AHALF1 to calculate the correction. Two alternate methods are provided for calculating CLAME if the End Item motor has a contoured nozzle. The approximate method uses the following equation above Statement 30.

$$CLAME = 0.5 + 0.5 * \cos((ALPHA + QEX) / 2.0)$$

In this equation ALPHA and QEX are determined from the contoured nozzle geometry as defined in paragraph 4.1.2. The method of characteristics scheme for calculating CLAME uses the following equation above Statement 30.

$$CLAME = (0.5 + 0.5 * \cos(ALPHA)) * XLAMBD$$

In this equation ALPHA and XLAMBD are those defined or calculated in paragraph 4.1.2.

The motor efficiency difference correction ETANZ/ETANZ1 is developed with the following second order curve fit equations. These equations are based on a semi-log plot of motor efficiency versus motor mass flowrates when the mass flowrate covers the range for both the test motor and the End Item motor.

$$ETANZ = (AK(10) + AK(11) * \text{ALOG}(WDE) + AK(12) * \text{ALOG}(WDE)**2) * SILSB$$

and

$$ETANZ1 = AK(10) + AK(11) * \text{ALOG}(WD2) + AK(12) * \text{ALOG}(WD2)**2$$

The curve fit coefficients are AK(10), AK(11) and AK(12). SILSB is the

6.1 (Continued)

empirically derived nozzle submergence loss factor. This factor must be developed with data from tests specifically designed to obtain the effect of nozzle submergence. The End Item motor mass flowrate (WDE) is calculated at Statement 20 with the following equation that relates characteristic exhaust velocity and the mass flowrate through the nozzle.

$$WDE = PCAVE * DTE ** 2 * CK(4) * CK(1) / (4.0 * MC(23))$$

In this equation PCAVE is the End Item motor average nozzle throat total pressure. The constants CK(4) and CK(1) are π (3.14159) and the gravitational constant, g(32.174 FT/SEC). The estimated End Item motor characteristic exhaust velocity is MC(23). The value of MC(23) is assumed to be the same as the test motor. The test motor mass flowrate WD2 is assumed to be available from test data and is a required input.

The End Item motor characteristic exhaust velocity CSTAR is calculated with the following equation.

$$CSTAR = MC(23) * OK(2) * ETANZ / ETANZ1$$

In this equation the test motor characteristic exhaust velocity MC(23) is corrected by the motor efficiency factor ETANZ/ETANZ1. An empirical factor OK(2) is provided in the event it is needed to further adjust motor efficiency.

The use of CSTAR in other modules requires CSTAR as a function of nozzle throat total pressure. Two options are provided for developing these data. The second order polynomial given in Statement 130 is used if Subroutine LEWIS was not used earlier in the Module. In this case, the curve fit coefficients CSCOEf must be input. If Subroutine LEWIS has been used it has calculated several additional theoretical characteristics exhaust velocities for a range of nozzle throat total pressures. Beginning at Statement 140, a routine is used to calculate a second order curve fit for these data to establish a theoretical characteristic exhaust velocity at the zero pressure intercept. This curve fit is performed by Subroutine LESSQ. The calculated zero pressure intercept is shifted so that the curve fit will pass through the End Item motor characteristic exhaust velocity, CSTAR. The revised intercept and the curve fit constants are set up in an equation for CSTAR as a function of End Item motor average nozzle throat total pressure as follows.

$$CSCOEf(1) = CSTAR - CSCOEf(2) * PCAVE - CSCOEf(3) * (PCAVE ** 2)$$

The data calculated in the Contractor Data Isp Module are transferred to the Output Module where they are printed in a specified format in both English and Metric Units.

6.2 CONTRACTOR DATA ISP MODULE LIMITATIONS

In order to scale test motor data to the End Item motor with the Contractor Data Isp Module it is necessary that the test motor and End Item motor have identical propellants and nozzle throat total pressures.

SECTION 7

7.0 INTERNAL BALLISTICS MODULE (IBM)

The Internal Ballistics Module (IBM) is used to calculate the internal ballistics performance of the End Item motor. Performance parameters calculated by IBM include vacuum thrust, total impulse, chamber pressures, moments of inertia, propellant center of gravity, burn surface area, and weight of propellant remaining. Each of these are printed at selected time intervals. These performance parameters are calculated based on nozzle and case geometry, propellant grain configuration and propellant characteristics. Propellant characteristics include density, specific heat ratio, gas constant, molecular weight, coefficient and exponent for the burn rate equation and characteristic exhaust velocity. Propellant characteristic exhaust velocity can be calculated with other Modules or input.

IBM is based on the Boeing Internal Ballistics Program described in Reference 6 and retains all the original capability plus the following additional capability.

- a) The grain geometry dividing planes have been increased to 18.
- b) Grain segments can be inhibited on one end.
- c) Burnrate can be varied as a function of radial burn distance.
- d) Nozzle throat erosion is a function of nozzle stagnation pressure.
- e) Nozzle efficiency is a function of time.
- f) Ambient pressure is a function of time.

In addition the original Boeing Internal Ballistics Program has been improved in the following areas.

- a) Buildup and decay performance calculation methods.
- b) Variable gas properties are interpolated with a linear interpolation routine.

7.1 INTERNAL BALLISTICS MODULE ENGINEERING DESCRIPTION

An engineering description of the Boeing Internal Ballistics Program is given in Boeing Document D2-125286-1, Solid Propellant Rocket Motor Internal Ballistic Computer Program - Program Manual (Reference 6). The Appendix included in this volume gives the necessary modified pages to revise Reference 6 in order to use it as an engineering description of the Internal Ballistics Module that is used in SRB-II.

7.2 INTERNAL BALLISTICS MODULE LIMITATIONS

The Internal Ballistics Module has the following limitations:

7.2 (Continued)

1. A maximum of 18 reference planes are allowed in the cylindrical section to describe the grain design.
2. A maximum of 100 increment dividing planes are allowed in the cylindrical section to define the mass addition regions. If this restriction is exceeded by defining a ΔZ too small, the case execution will be terminated and an appropriate comment will be printed.
3. A maximum of 18 slots are allowed for segmented motors.
4. The slotted-cone grain design is applicable only to the cylindrical section. Burn area tables must be input for the forward and aft domes when this grain design is used.
5. The inert sliver option is restricted to the cylindrical section and does not apply to either end section.
6. The effects of an accelerating reference system can be determined only for non-steady flow gas dynamics.
7. The submerged nozzle entrance plane must be perpendicular to the nozzle centerline.
8. Uninhibited or fully inhibited segment faces can be simulated; however, partially inhibited segment faces cannot be simulated. Partially inhibiting occurs when a lateral face of a segment is only partially covered with burn inhibiting material.

SECTION 8

8.0 THRUST SCALING MODULE (FSCAL)

The Thrust Scaling Module provides a method for calculating nominal End Item motor performance by scaling the performance of a baseline motor. The procedure produces a scaled thrust versus time trace from a similar trace of the baseline motor by modifying total burn time or total propellant mass. The module was designed for an alternate to the Internal Ballistics Module when data needed for a complete internal ballistics prediction are not available.

The Module uses the Subroutines given in Table 8-I.

TABLE 8-I: THRUST SCALING MODULE SUBROUTINES

<u>Subroutine Name</u>	<u>Paragraph Number</u>
INERT	9.0
OUTPUT	12.0

Each of these Subroutines are described in other paragraphs and will not be repeated in this section.

8.1 THRUST SCALING MODULE ENGINEERING DESCRIPTION

The first part of the module, down to above Statement 10, sizes and locates the Real and Common variables used in the module. NPMAX is set equal to the number of points used in the baseline thrust (AFSRM(I)) versus time (ATB(I)) data given in Table 3-XIV of Volume II.

Statements 10 through 30 set the values of SRB burn time TB and TBNOM and propellant mass (MPTOT) depending on the values of the indicators NTB and NMP.

Beginning at Statement 30 several calculation variables are initialized. These include the scaled time (TPN), scaled SRB thrust (FSRM), scaled SRM flowrate (MFSRM), scaled SRB propellant mass remaining (MPR(I)) and scaled SRB chamber pressure (PC(I)).

The performance scaling calculations begin at Statement 40 with a loop counter that sets up each baseline thrust versus time point that is to be scaled. The baseline motor data are given in Table 3-XIV of Volume II. The following paragraphs will describe each scaling equation used in the loop.

The time increment used in the baseline thrust versus time table is scaled to the nominal time increment (TPN(I)) with the following equation.

$$TPN(I) = TB / TBNOM * ATB(I)$$

8.1 (Continued)

In this equation TB is the burn time for the scaled motor, TBNOM is the burn time for the baseline motor and ATB(I) is the time increment used in the baseline motor thrust time table.

The value of baseline thrust in the thrust versus time table are scaled with the following equation.

$$FSRM(I) = TBNOM/TB * MPTOT/MPNOM * AFSRM(I)$$

In this equation TBNOM and TB are defined above. AFSRM(I) is the baseline thrust given in the baseline motor thrust time table. The baseline motor propellant mass MPNOM is consistent with the data in the thrust versus time table. MPTOT is the propellant mass of the scaled motor.

The scaled motor flowrate (MFSRM(I)) is calculated with the following equation:

$$MFSRM(I) = FSRM(I)/SIDE$$

In this equation the scaled motor thrust FSRM(I) was calculated above and SIDE is the scaled motor average specific impulse.

The propellant mass remaining (MPR(I)) in the scaled motor is calculated with the following equation.

$$MPR(I) = MPR(I-1) - (MFSRM(I-1) + MFSRM(I))/2.0 * DELT$$

In this equation MFSRM(I) is calculated above. The time increment (DELT) is the time between the scaled thrust versus time data.

The chamber pressure of the scaled motor is calculated with the following equation.

$$PC(I) = MC(1) + FSRM(I)$$

This equation is derived from the standard equation relating thrust, chamber pressure, throat area and coefficient. The scaled motor thrust (FSRM(I)) was calculated above and (MC(1)) is the inverse of the product of the scaled motor thrust coefficient and nozzle throat area.

Subroutine INERT is used to calculate the contribution of the "inert" thrust, "inert" flowrate and "inert" mass remaining on board. These data are scaled from a baseline data of "inert" mass overboard versus propellant mass overboard (Table 3-XIV of Volume II). These data are nondimensionalized so that the total propellant and "inert" mass may be varied to match the scaled motor.

8.1 (Continued)

The data calculated in the Thrust Scaling Module and the data calculated in Subroutine INERT are transferred to the Output Module where they are combined and printed in both English and Metric Units.

8.2 THRUST SCALING MODULE LIMITATIONS

The thrust scaling technique used in the Thrust Scaling Module is restricted to small deviations ($\approx 5\%$) from the thrust versus time data for the baseline motor. Use of the Module also requires that the scaled motor have the same thrust versus time shape as the baseline motor.

SECTION 9

9.0 "INERT" MASS MODULE (INERT)

The "Inert" Mass Module calculates inert mass flow rate, mass of expendable inerts remaining, thrust contribution of the inerts, and the total impulse contribution of the inerts. Inert mass accounting is required because the total mass expended by a solid rocket motor exceeds the propellant mass. The excess is considered expended "inert" mass and is made up primarily of insulation and ablative material contained in the motor interior and nozzle throat. This expended mass has a slight influence on thrust and total impulse and a very significant influence on motor mass versus time.

When a thrust scaling prediction is being made, INERT is called by FSCAL. When an internal ballistics simulation is being made, INERT is accessed by the Internal Ballistics Module through subroutine FAMCAL.

9.1 "INERT" MASS MODULE ENGINEERING DESCRIPTION

The beginning statements of INERT up to Statement 10 establish the mass of propellant remaining, MPR, and the computation time increment, DELTIM, when an internal ballistics simulation is being made (i.e., when NF=1). Also, when TIME=0., the initial total mass of propellant, MPTOT, and the initial computation time increment, DELTIM, are set.

For a thrust scaling prediction (i.e., when NF=0) data for MPTOT and MPR are available through FSCAL and control is transferred to Statement 10 to initialize the computation time increment, DELTIM.

Starting with Statement 20, MPRR is calculated as the ratio of the current mass of propellant remaining to the initial mass of propellant. This is now the independent variable which is used to determine the dependent variable MIRR which is the corresponding ratio of the mass of inerts remaining to the initial total mass of inerts. MIRR is obtained with a linear interpolation of the AMPRR(I) and AMIRR(I) input array data.

After the ratio MIRR is determined, the mass of inerts remaining, MIR, is calculated as the product of MIRR and the initial total mass of inerts, MITOT. The inert mass flowrate, MIF, is then calculated as the change in mass over the computation time interval divided by the time interval. The consumed mass of inerts, MFINIT, is calculated by numerically integrating the flowrate. The inert thrust contribution, FI, is calculated as the product of the flowrate and the input value for the average inert mass specific impulse, SII. The inert contribution to the delivered total impulse and the vacuum total impulse (AITIN and AITVIN, respectively) are calculated as functions of thrust.

The variables whose names end in Z (MIZ, MIFZ, FIZ, and TIMEZ) are set to current values for use as previous values in the subsequent pass through the subroutine.

SECTION 10

10.0 RECONSTRUCTION MODULE (RECON)

The Reconstruction Module is used to analyze the propulsion system performance of SRB solid propellant rocket motors after the motor firing has been completed. Data from either static or flight firings may be used in conjunction with the Reconstruction Module to produce motor mass flowrate, mass overboard, and specific impulse versus time. Post-firing analysis of test data using the Reconstruction Module also produces parameters such as characteristic exhaust velocity and burning rate adjustments which can be used to improve accuracy of future predictions. The modules and subroutines called by the Reconstruction module are presented in Table 10-I along with paragraph numbers in which the routine engineering description is found.

TABLE 10-I: SUBROUTINES AND MODULES CALLED BY THE RECONSTRUCTION MODULE

<u>Subroutine Name</u>	<u>Paragraph Number</u>
BIAS	10.2
PADJ	10.3
IBM	7.0
LESSQ	4.5
LEWIS	4.3
LEWIT	4.3

Each of these subroutines except BIAS and PADJ have been described in other paragraphs and will not be repeated here.

10.1 RECONSTRUCTION MODULE ENGINEERING DESCRIPTION

The Reconstruction Module is based upon the following postulation: "For a set of pressure data from a firing of a motor with known propellant geometry and known nozzle geometry, there exists a unique relationship between propellant burning area, measured pressure, and the propellant burning rate law." The criterion for successful reconstruction of test performance using the Reconstruction Module is a match between measured head-end (forward-end) pressure and calculated burn area. This successful match must yield a burning rate law coefficient of pressure (a in a P^n) which is consistent with the linear burning rate law with augmentation ($r = a P^n[1 + \delta]$). The analysis implicitly assumes that the burning rate law pressure exponent, n , is known from propellant test data and does not vary during the firing. It is also assumed that the coefficient of pressure, a , is primarily a function of grain temperature for a well mixed propellant.

Analytical calculations of motor internal ballistics performance in a reconstruction are carried out in the Internal Ballistics Module (IBM). The mode of IBM exercised during reconstruction is the same as the mode used for a buildup transient calculation with tabular input of head-end pressure data. These calculations force a match between input head-end pressure and calculated burning area. One result of these calculations is a value of coefficient of pressure which forces the propellant burning

10.1 (Continued)

rate to satisfy the gas dynamic relations for mass conservation at each time slice. The values of coefficient of pressure derived at each time slice must change smoothly with time during a firing reconstruction. Any abrupt change in coefficient of pressure represents an inconsistency between calculated burning area and measured pressure. The cause of this inconsistency is an incorrect calculated propellant grain regression rate.

The net effect of this incorrect calculated regression rates over a firing are realized at times when web burn throughs accompanied by rapid changes in burning area occur. The measured head-end pressure transients which correspond to changes in burning area will be out of phase with the calculated burning area changes. Thus, when a match between measured head-end pressure and burning area is forced by IBM, the value of coefficient of pressure produced deviates from the established trend in accordance with the degree of inconsistency between measured pressure and calculated area. This deviation from the established trend is used in the Reconstruction Module to determine when a reconstruction has converged to a good match across the run.

Regression rates in the simulation are controlled by several input factors such as propellant characteristic exhaust velocity, propellant density and propellant geometry. In cases where the coefficient of pressure deviates from the data trend, the most suspect of these input parameters is the propellant characteristic exhaust velocity (c^*). For cases in which the input value of c^* is high, the calculated propellant grain regression rate will be low when compared with real values of regression. Thus the burning area will remain at a relatively high level at initialization of a measured pressure transient and the calculated value of coefficient of pressure is forced down, i.e.,

$$c^* \uparrow A_b \rightarrow P_m \downarrow \Rightarrow a \downarrow$$

Likewise, low characteristic exhaust velocity input causes high regression rates and

$$c^* \downarrow A_b \downarrow P_m \rightarrow \Rightarrow a \uparrow$$

The Reconstruction Module manipulates the characteristic velocity curve fit up or down according to the direction in which coefficient of pressure deviates from the data trend.

Recognition of coefficient of pressure deviations is achieved by first utilizing statistical techniques to establish a data perturbation band about the data trend line at each time slice, and then checking for perturbations about the data trend which are beyond this limiting band. Data which deviate outside the perturbation band cause the routine to re-evaluate the characteristic exhaust velocity curve fit and re-initialize the reconstruction case at a predetermined time.

10.1 (Continued)

The first part of the Reconstruction Module (RECØN) sizes and locates Real, Logical and Common variables used in the module. The logical variable LIST1 is tested. If LIST1 is true, the following LIST1 variables were input to the routine:

- TSREC - Time, referenced to ignition time, at which measured head-end pressure is equal to PF AG during buildup.
- TEREC - Time, referenced to ignition time, at which measured head-end pressure is equal to PFLAG during tailoff.
- CSBAR - Characteristic velocity at pressure PBAR, used to shift the theoretical characteristic exhaust velocity versus pressure curve.
- PBAR - Pressure used to shift the theoretical characteristic exhaust velocity versus pressure curve.
- TWEB - Web time of the grain, defined as the time at which a specified percentage (PHEPI) of the head-end pressure integral is reached.
- PITW - Numerical value of head-end pressure integral used to define the time TWEB. PITW equals total head-end pressure integral multiplied by the specified percentage (PHEPI) for TWEB.

If the variable LIST1 is false, values for the LIST1 variables are computed between the test on LIST 1 and Statement number 70. The variables TIMEZ, PHEADZ, TERC, TWEB, PITW, PNSINT, and PHINT are initialized to zero. Next the reconstruction data tape is rewound after Statement 10 and one record is read from the tape after Statement 15. Subroutine BIAS is called to apply any required zero corrections and bias factors to the measured data. After the call to BIAS a series of tests is performed to establish values of TSREC and TERC. These tests continue to Statement 40. At Statement 40 the nozzle stagnation pressure, PNS, is calculated from measured head-end pressure and an adjustment function PADJ(TIME). Next, the integrals of head-end pressure and nozzle stagnation pressure are formed using a trapezoidal rule integration scheme.

After the pressure integrations, a test for the end of the reconstruction data is performed. If TIME is greater than or equal to TERC, the program logic branches to Statement 50. If the test on time fails, a test to determine whether or not a value of PITW has been calculated is made. If no value has been computed for PITW, previous values of integration parameters are set to current values at Statement 42 and the program logic branches to Statement 15 for the next data record from tape. If a value of PITW has been calculated, program logic branches to Statement 45 where

10.1 (Continued)

a test to determine whether the integral of head-end pressure, PHINT, is less than PITW. If PHINT is less than PITW, program logic branches to Statement 42. If PHINT is not less than PITW, the value of TWEB is calculated using linear interpolation on time and the program logic branches to statement 70.

At Statement 50 a test for TWEB greater than zero is made. If the test is passed, program control branches to Statement 70. If the test is failed, values for PITW, CSBAR and PBAR are calculated.

Next a test is made to determine if there is a requirement to execute Subroutine LEWIS. If NLEWIS is greater than zero, Subroutine LEWIS is to be called. Set-up for and calls to LEWIT and LEWIS are executed between Statement 55 and the DO 58 I=1, NPTS statement. Pressures are converted from points per square inch to atmospheres in the Subroutine LEWIS and are converted back to pounds per square inch in the DO 58 loop.

After Statement 58 Subroutine LESSQ is called to fit a second order least squares curve fit through the theoretical characteristic exhaust velocity as a function of pressure data generated in Subroutine LEWIS. Next a test is executed to determine whether TWEB has been calculated. If TWEB has been calculated, program logic branches to Statement 75. If TWEB has not been calculated, it is necessary to reintegrate the pressure versus time curve in order to determine when TWEB occurs; this procedure requires the program logic to branch to Statement 10 to rewind the data tape and restart the integration.

Statement 70 is the starting point for the initial pass through the matching loop in a reconstruction run. At Statement 70 a test is made to determine whether a call to Subroutine LEWIS is required. If NLEWIS is greater than zero, the program logic branches to Statement 55 and Subroutine LEWIS is called to provide theoretical characteristic exhaust velocity versus pressure data as previously described. Once the LEWIS data are available, the curve fit of the theoretical characteristic exhaust velocity data is shifted such that it passes through the point (CSBAR, PBAR) at Statement 75.

Statement 85 is the main branch point in the iterative reconstruction convergence loop. Each iteration is initiated at Statement 85. After Statement 85 IBM is called to calculate the time varying parameters and to set the reconstruction convergence indicator AFIT as described in Section 9.1.1 of this volume. After the call to IBM, a test is performed to determine whether the current iteration is the first iteration. If IPASS is greater than zero, the current iteration is not the first iteration and program logic branches to Statement 110. If IPASS is equal to zero, the loop counter (ILOOP), the delta characteristic exhaust velocity for iteration (DELCS) and the characteristic exhaust velocity direction indicator (CSIND) are initialized and IPASS is set to one.

10.1 (Continued)

At Statement 110, the loop counter is incremented. Next, two tests are made to determine whether the reconstruction converged during the IBM simulation. The first test checks for reconstruction divergence which caused the coefficient of pressure to deviate from the established data trend in a positive direction (AFIT is greater than zero). If this test is passed, the assumed characteristic exhaust velocity curve fit was low and the program logic branches to Statement 130.

The second test checks for reconstruction divergence which caused the coefficient of pressure to deviate from the established data trend in a negative direction (AFIT is less than zero). If this test is passed, the assumed characteristic exhaust velocity curve fit was high and the program logic branches to Statement 140. Reconstruction convergence is achieved when both tests fail (AFIT equal to zero) and the case is terminated with a successful completion message printed from Format 1000.

Statements 130 and 140 test on the loop counter for maximum number of iterations equaled or exceeded. If the test is passed the program logic branches to Statement 120 where a non-convergence message is written from Format 1010 and the case is terminated. If the maximum number of iterations has not been executed, the characteristic exhaust velocity curve fit is adjusted up or down as necessary by changing the zero pressure curve intercept CSCOE(1) appropriately. The program logic then branches to Statement 150 where a header message is printed from Format 1020 and the iteration loop is reinitiated.

10.1.1 Reconstruction Calculations in IBM Subroutine MNCHN4

A portion of the calculations required to determine whether convergence has been achieved in reconstruction is located in IBM Subroutine MNCHN4. These calculations are located between Statements 1270 and 1278. Calculations are performed in Subroutine MNCHN4 to increase speed and efficiency of computation.

Subroutine MNCHN4 calculations produce a least squares second order curve fit of coefficient of pressure versus distance burned using data from the four previous computation cycles. Next a value of coefficient of pressure (APROJ) is projected for the current value of distance burned and the difference between the current calculated value and the projected value of coefficient of pressure (ADEL) is computed. The standard deviation of the previous four coefficient of pressure points is calculated in Subroutine VARI.

ADEL is compared with the value of three standard deviations for the previous four points (A3SIG). If the absolute value of ADEL is greater than A3SIG, the coefficient of pressure is considered to have deviated from the data trend and the indicator AFIT is set to indicate the direction of deviation. Next a message is written to indicate the deviation and the program branches to Statement 1278. If the coefficient

10.1.1 (Continued)

of pressure has not deviated from the data trend, the tables are updated for the next computation cycle and program logic branches to Statement 1278.

10.2 SUBROUTINE BIAS ENGINEERING DESCRIPTION

Subroutine BIAS is used to algebraically add bias factors such as zero correction and drift correction to measured data. Bias factors are installed at the time of reconstruction. Installation of bias factors requires recompilation of Subroutine BIAS.

10.3 FUNCTION PADJ(TIME) ENGINEERING DESCRIPTION

Function PADJ is used to provide an estimate of pressure drop from head-end pressure to nozzle stagnation pressure. The change in pressure drop is calculated as a linear function of time. The slope and intercept for the pressure adjustment can be derived from the pre-firing prediction for the motor configuration.

10.4 RECONSTRUCTION MODULE LIMITATIONS

MNCHN4 reconstruction logic is set up to operate on data from mass addition region one. For configurations in which mass addition region one burns laterally as well as radially, another mass addition region which burns only the radial direction should be chosen and data from this chosen region should be substituted for the region one data. Substitution of another region will require recompilation of Subroutine MNCHN4.

Other limitations for the Reconstruction Module are the same as those listed for the Internal Ballistics Module (Paragraph 7.2).

SECTION 11

11.0 DISPERSION MODULE

The Dispersion Module perturbs the independent input variables for a dispersion data analysis. The Dispersion Module intercepts the input data for the Internal Ballistics Module (IBM) by reading the IBDATA NAMELIST for the nominal case (first case input for a run), modifies these data to yield the required dispersion in the desired parameter, and writes the dispersed data on disk in NAMELIST IDISP. (NOTE: Input data requirements and the input philosophy are discussed in Volume II of this document). The IBM then reads the data from IDISP NAMELIST and proceeds in the same manner as a nominal prediction. The Dispersion Module calls no subroutines or modules other than the Output Module for preliminary printout of dispersion data.

11.1 DISPERSION MODULE ENGINEERING DESCRIPTION

The module initializes by defining the I/O unit numbers for the IBDATA NAMELIST (NIBOUT=20) and rewinding the I/O unit for the IDISP NAMELIST (REWIND 25). Next a series of variable initialization statements are executed in which the variable SETFLG is set to zero and the variable MITSAV is set to MITOT. This action is required to allow execution of multiple cases in which an "inert mass" dispersion is not the final case.

A test is performed to check for a thrust scaling prediction and in the thrust scaling case, IBM variable initialization is skipped. For an IBM prediction, several IBM input variables are then set to zero by the ensuing statements. This is done to initialize these variables such that in the case that no value is input by the user for any one or all of these variables, useless data is not written out in the IDISP NAMELIST and used by the IBM.

The IBDATA NAMELIST I/O unit (NIBOUT) is rewound and IBDATA is read. The Output Module is called to give preliminary dispersion data printout of the Dispersion Option List. After the call to Output a check to determine whether an "inert mass" dispersion was executed in the previous case (SETFLG=1) is executed. If an "inert mass" prediction was run in the previous case, the "inert mass" initial value is set back to the nominal value.

At Statement 15 the module selects the method for the specific dispersion calculation that will be made based on the value of NDISP. The dispersion options available in the module are given in Table 11-1.

In the case where NDISP=1, the module goes to Statement 20 where it perturbs the nominal value of the propellant density (DELFL) by the dispersion value for propellant density (DISLIM(1)). The perturbed value for (DELFL) is printed to provide a label identifying the specific dispersion ahead of the normal print format. The perturbed value of DELFL replaces the nominal value of DELFL and is stored on disk in NAMELIST IDISP and will be input to the Internal Ballistics Module for calculating the booster performance with the perturbed propellant density.

11.1 (Continued)

TABLE 11-I: SRB DISPERSION VARIABLES

NDISP= 1 :	+ PROPELLANT DENSITY
NDISP= 2 :	- PROPELLANT DENSITY
NDISP= 3 :	+ PRESSURE EXPONENT
NDISP= 4 :	- PRESSURE EXPONENT
NDISP= 5 :	+ PRESSURE COEFFICIENT
NDISP= 6 :	- PRESSURE COEFFICIENT
NDISP= 7 :	+ CHARACTERISTIC VELOCITY
NDISP= 8 :	- CHARACTERISTIC VELOCITY
NDISP= 9 :	+ PROPELLANT GRAIN LENGTH
NDISP=10 :	- PROPELLANT GRAIN LENGTH
NDISP=11 :	+ PROPELLANT GRAIN WEB THICKNESS
NDISP=12 :	- PROPELLANT GRAIN WEB THICKNESS
NDISP=13 :	+ INITIAL THROAT DIAMETER
NDISP=14 :	- INITIAL THROAT DIAMETER
NDISP=15 :	+ INITIAL EXIT DIAMETER
NDISP=16 :	- INITIAL EXIT DIAMETER
NDISP=17 :	+ THROAT EROSION RATE
NDISP=18 :	- THROAT EROSION RATE
NDISP=19 :	+ PROPELLANT GRAIN TEMPERATURE
NDISP=20 :	- PROPELLANT GRAIN TEMPERATURE
NDISP=21 :	+ INITIAL INERT MASS CONSUMABLE
NDISP=22 :	- INITIAL INERT MASS CONSUMABLE

The procedure described in the proceeding paragraphs can be repeated for each of the performance parameters given in Table 11-I.

SECTION 12

12.0 OUTPUT MODULE

The Output Module is capable of producing data in the following three forms: tabulated computer printout, seven and nine track-performance data tapes (UNIVAC 1108 version generates seven track tapes while IBM/370 version generates nine track tapes) and plot tapes, and punched data cards. The last two items are optional outputs and their generation are determined by setting input indicators (NPLOT, NTAPE, or NCARD). The Output Module uses the Subroutines given in Table 12-1.

TABLE 12-1: OUTPUT MODULE SUBROUTINES

<u>Subroutine Name</u>	<u>Paragraph No.</u>
PRT1	12.1
PRT2	12.1
PRT3	12.1
PRT4	12.1
PRT5	12.1
PRT6	12.1
PRT7	12.1
PRT8	12.1

Each of these Subroutines will be described as a part of the Output Module.

12.1 OUTPUT MODULE ENGINEERING DESCRIPTION

This subroutine controls the printing of the module data printouts. The first 3 statements after the subroutine name assigns the variable locations and type. The computed go to statement, before Statement 100, controls branching to appropriate module print subroutine based on the value of NPRINT. NPRINT is set in the Executive Module (MAIN) before each module call statement. Statements 100 through 1000 are the calls to the various print subroutines and the branch statements to return program control to the calling routine. The one statement NL600=50, after Statement 100, is to initial the number of lines counter used in subroutine PRT6. The following is a description of the function of Subroutines PRT1 through PRT8.

- PRT1 - Prints the initial values for a case. They include program, motor, curve fit, universal and odd constants. They also include general program indicators, optional print indicators, and inert mass data.
- PRT2 - Prints the data calculated in the Specific Impulse Scaling Module (SISCAL) in both metric and english units.
- PRT3 - Prints the data calculated in the Contractor Data Specific Impulse Module (CDSI).

12.1 (Continued)

- PRT4 - Prints the data calculated in the BATES Specific Impulse Module (BATES).
- PRT5 - Prints the dispersion values for a case (DISP).
- PRT6 - Prints the data calculated in the Thrust Scaling Module (FSCAL).
- PRT7 - Prints the data calculated in the Internal Ballistics Module (IBM).
- PRT8 - Prints specific data from the Reconstruction Module (RECON).

SECTION 13

13.0 NOZZLE SUBMERGENCE AND CONTOUR EFFECTS MODULE (NSCE)

The Nozzle Submergence and Contour Effects Module (NSCE) incorporates a mathematical model to simulate the effects of nozzle contour and submergence on the internal ballistics and overall performance of a solid propellant rocket motor. Contour effects are simulated using functional modeling techniques which account for changes in performance due to nozzle expansion section shape. Since the primary effect of nozzle contouring is a change in the divergence loss factor, NSCE derives a value for this factor which allows the Internal Ballistics Module (IBM) to get the proper divergence loss factor for the contoured nozzle without change to the original IBM calculation technique.

Submergence effects are modeled using both analytical and functional techniques to account for changes in the motor internal flow field and performance. The NSCE Module simulates the internal ballistics in the motor aft-end when the nozzle is submerged. For a submerged nozzle simulation, NSCE replaces IBM subroutines AIBSUB and AIBST for calculation of internal gas dynamics in the aft-end of the motor. The simulation differs from a normal internal ballistics simulation in that reversed flow (flow toward the forward end) occurs in the submerged portion of the port and more than one stagnation region occurs as shown in Figure 13-1.

The assumptions made in deriving the nozzle submergence simulation in the NSCE Module are consistent with those made in the derivation of the IBM aft-end simulation with the additional specification that the nozzle entrance and submerged section be centered in the port for all cases. The basic assumptions for IBM and NSCE are as follows:

- (a) The mass addition occurs as an instantaneous process with no velocity component parallel to the motor axis.
- (b) The products of combustion obey the perfect gas law.
- (c) The gas flow is one-dimensional and adiabatic.
- (d) The friction forces of the combustion gases in the port cavity are negligible.
- (e) The static pressure is constant across the fore-head section, i.e. no static pressure loss resulting from mass addition or area change.

The Subroutines called by NSCE are listed in Table 13-I. Descriptions for Subroutines RSUB and XLIN are presented in preceeding paragraphs and will not be repeated.

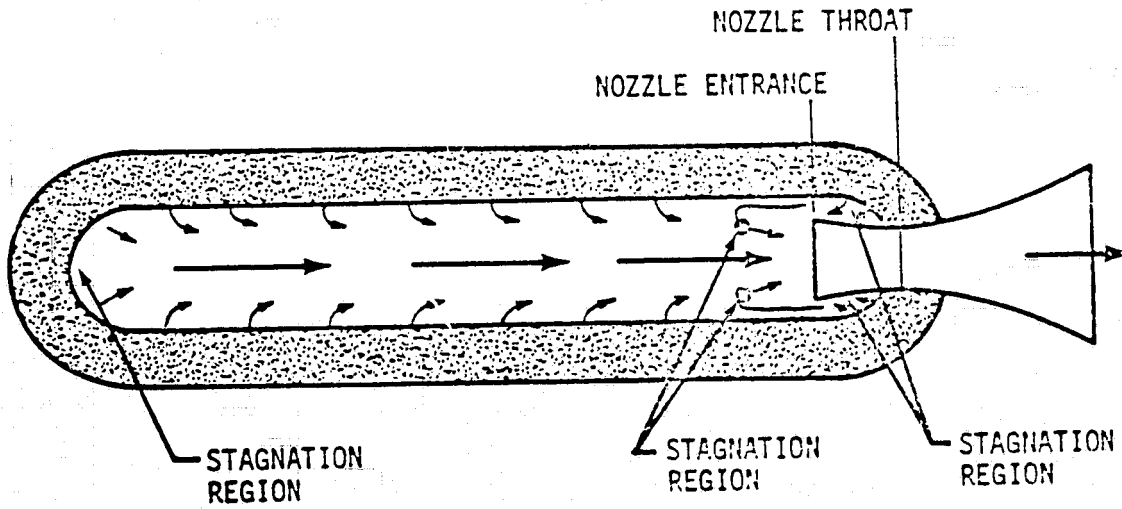


FIGURE 13-1 INTERNAL FLOW WITH A SUBMERGED NOZZLE

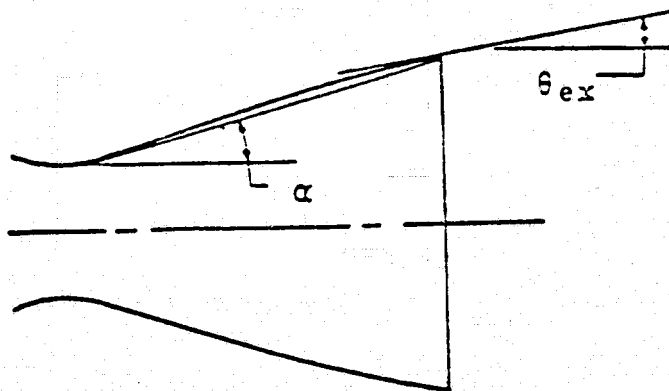


FIGURE 13-2 DEFINITIONS OF THE ANGLES α AND θ_{ex} FOR A CONTOUR NOZZLE

13.0 (Continued)

TABLE 13-I: NSCE MODULE SUBROUTINES

<u>Subroutine Name</u>	<u>Paragraph Number</u>
MACH	13.2
RBSUB	7.0
SUBSON	13.3
XLIN	7.0

13.1 NOZZLE SUBMERGENCE AND CONTOUR EFFECTS MODULE ENGINEERING DESCRIPTION

The NSCE Module is accessed only from the IBM. An internal option indicator, IOPT, is set in the calling statement which directs the NSCE Module to do the following:

- IOPT=1 Correct the nozzle half angle for nozzle contour effects as described in paragraph 13.1.1.
- IOPT=2 Perform an internal ballistics simulation of the aft-end section of the motor for a specified submerged nozzle section assuming steady state gas dynamics apply. This simulation is discussed in paragraph 13.1.2.1.
- IOPT=3 Perform an internal ballistics simulation of the aft-end section of the motor for a specified submerged nozzle section assuming non-steady state gas dynamics apply. This simulation is discussed in paragraph 13.1.2.2.

After any one of these options is exercised, the module returns to the calling routine. Thus, the NSCE Module is divided into three separate sections which are independent.

13.1.1 Contour Effects Simulation (IOPT=1)

For cases in which the motor to be simulated has a contoured nozzle expansion section, a functional simulation routine is executed to provide the IBM with a proper datum for thrust coefficient calculation. This simulation is performed between statements 10 and 100.

The primary effect of nozzle contouring is a change in the divergence loss factor λ . The divergence loss factor is calculated in IBM from the effective half angle for the nozzle. Thus the IBM datum is an effective half angle for the nozzle. The IBM delivered thrust equations are as follows:

$$C_{FOL} = (C_{FO} \lambda_n + \frac{P_E}{P_{ON}} \epsilon_G) C_M$$

13.1.1 (Continued)

$$FDEL = C_{FOL} P_{ON} A_t - P_a \epsilon_G A_t$$

where C_{FOL} = Momentum portion of the thrust coefficient

$$\frac{P_E}{P_{ON}} = \text{Exit pressure ratio}$$

ϵ_G = Nozzle exit expansion ratio

C_M = Thrust coefficient efficiency

A_t = Nozzle Throat Area

P_a = Ambient Pressure

λ_n = Divergence loss factor

(NOTE: Vacuum thrust is $FVAC = C_{FOL} P_{ON} A_t$).

The divergence loss factor is calculated from

$$\lambda_n = .5 + .5 \cos(AN2)$$

where AN2 is the effective conical nozzle half angle.

Two methods of calculation of AN2 are available in the NSCE Module. The first is an approximate method which calculates AN2 as the average of the effective cone angle for the contour (α) and the contour exit angle (θ_{ex}) as recommended in Reference 7. The angles α and θ_{ex} are defined in Figure 13-2. The angle α is input as AN2 in the IBDATA NAMELIST AND θ_{ex} is input as THETEX. The approximate calculation of AN2 for a contoured nozzle is made in the following statement:

$$IF(THETEX.GT.0.0)AN2=(AN2+THETEX)/2.0$$

The second method of calculation for AN2 requires theoretical specific impulse data from a more sophisticated analysis of nozzle performance. The ratio of the contoured nozzle theoretical specific impulse to the theoretical specific impulse for an equivalent conical nozzle is input as SIRATC. The half angle for the equivalent cone is input as AN2 and the contoured nozzle effective angle is calculated by the following statements:

$$\begin{aligned} AN2R &= PI * AN2 / 180. \\ AN2R &= ACOS(SIRATC * (COS(AN2R) + 1.0) - 1.0) \\ AN2 &= 180. * AN2R / PI \end{aligned}$$

13.1.1 (Continued)

It should be noted that either the first or the second method of AN2 calculation may be applied, but not both. If a value is input for both THETEX and SIRATC, the program writes an error message and terminates the case. If a value for neither is input, the case also terminates with an error message.

13.1.2 Nozzle Submergence Effects Simulation (IOPT=2 and 3)

Both the steady state (IOPT=2) and non-steady state simulations (IOPT=3) use the same basic geometric configuration. This configuration assumes two mass addition regions in the aft end as shown in Figure 13-3. Definitions for mass addition region A are shown in Figure 13-4 and definitions for region B are shown in Figure 13-5.

NSCE gas dynamic theory is based on the premise that the overall computational scheme of IBM is not modified. This premise requires that the solution to the gas dynamic equations for regions A and B not force the mass flowrate out of region A to match the calculated nozzle mass flowrate. Hence, conditions in the nozzle entrance plane are not computed from choked flow rate and subsonic nozzle flowrate calculations, but are determined from the isentropic Mach number relationships, the continuity equation and the momentum equation. These equations, as well as the basic equations for region B, are iterated using a master iteration scheme until a consistent set of values is obtained.

The master iteration scheme for the solution to the submerged nozzle gas dynamic equations for both regions A and B is shown in Figure 13-6. In both steady and non-steady simulations, the conditions at the forward end of region B are calculated for some assumed stagnation pressure in the aft-end of region B. The conditions in the nozzle entrance portion of region A are then calculated. A comparison of the total pressures calculated for the forward end of region B and the nozzle entrance portion of region A is made. If the total pressures are equal within a tolerance of 0.01 percent, the solution is complete. If the total pressures are not equal within tolerance, a new stagnation pressure in the aft end of region B is computed using a modified slope prediction scheme and the iteration is continued until either the total pressures match or an iteration limit is exceeded. For cases in which the iteration limit is exceeded, an error message is printed and the case is terminated. A total pressure match causes an end to NSCE calculations and returns program control to the calling routine.

Parameter initialization is performed between statements 100 and 110. These parameters include conditions at the entrance to region A (total pressure PIAT and static pressure P1A), iteration loop counters (ITP1B, ITP2B, ITT1B), area parameters (AENT, AR, A2A), and combustion products properties data. Constant combustion products properties are assumed in the variable NCSTR is not greater than 0.0. For NCSTR greater than 0.0,

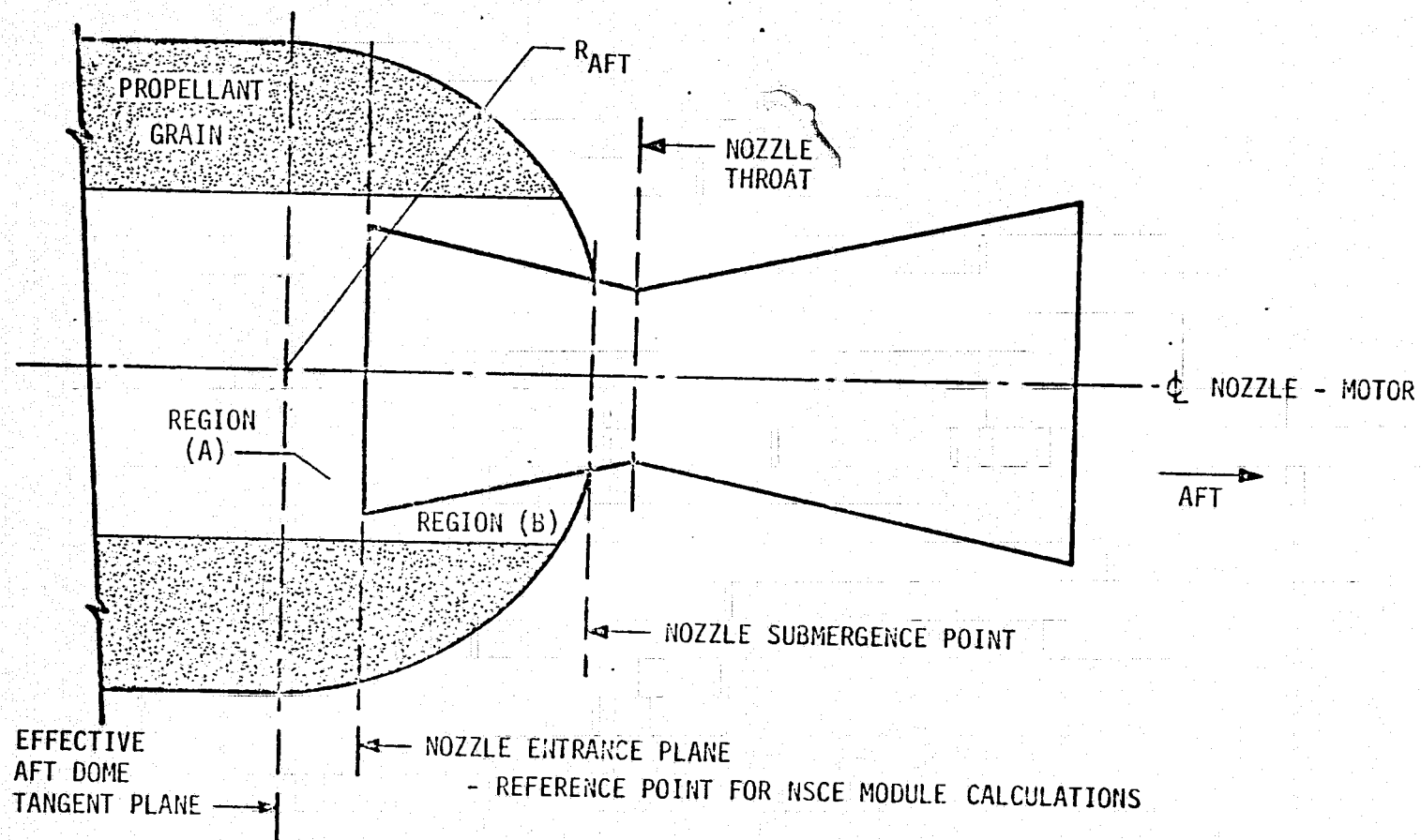


FIGURE 13-3 ASSUMED AFT-END CONFIGURATION WITH A SUBMERGED NOZZLE

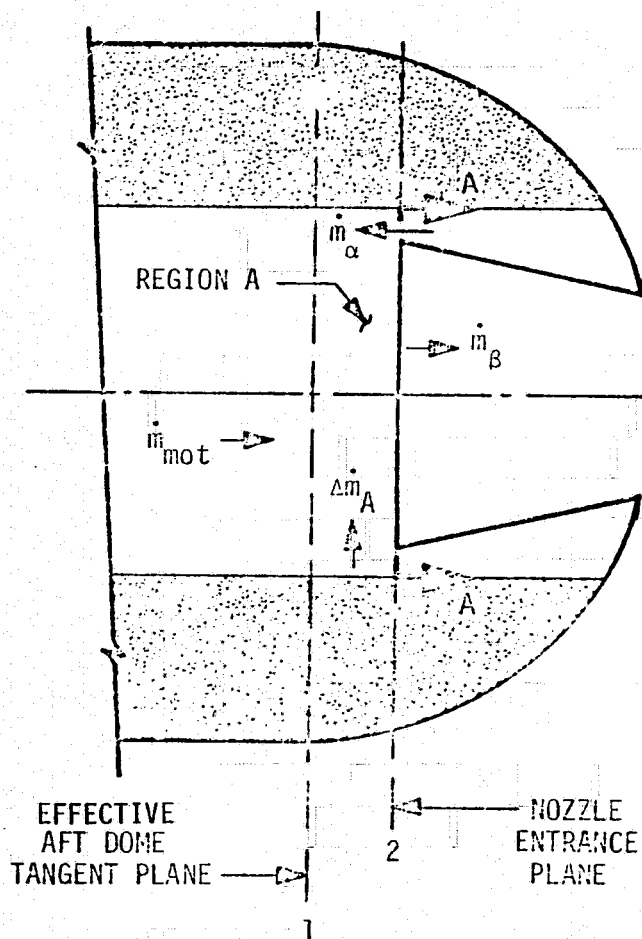
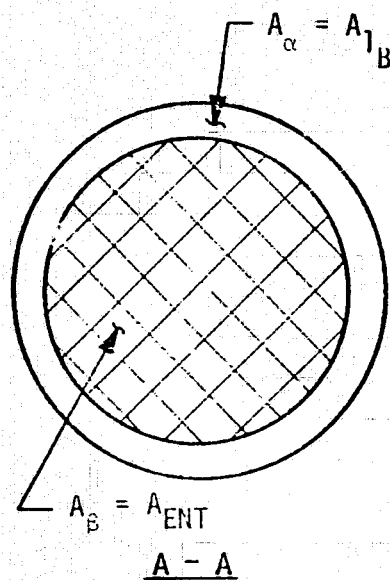


FIGURE 13-4 DEFINITIONS FOR MASS ADDITION REGION A

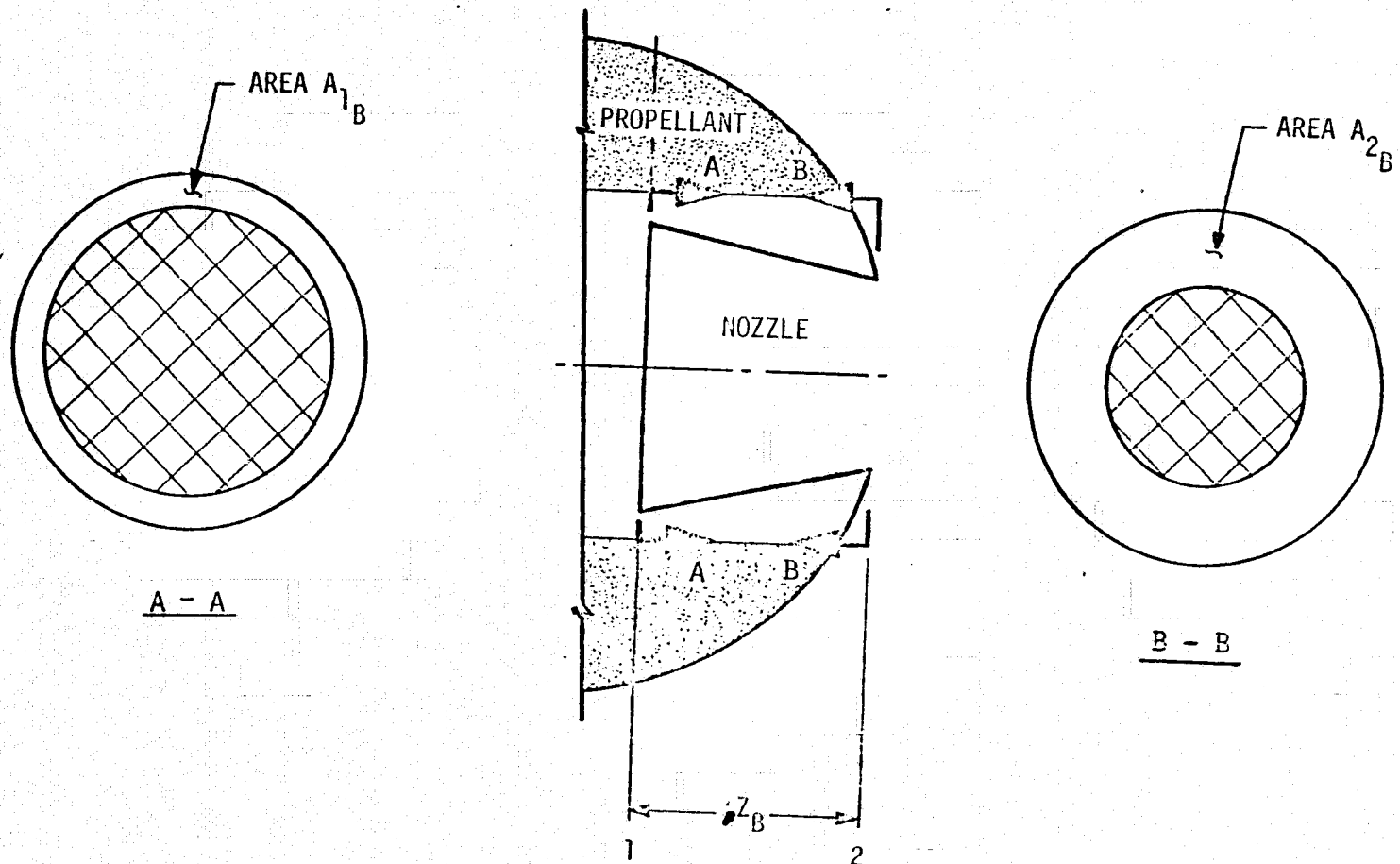


FIGURE 13-5 DEFINITIONS FOR MASS ADDITION REGION B

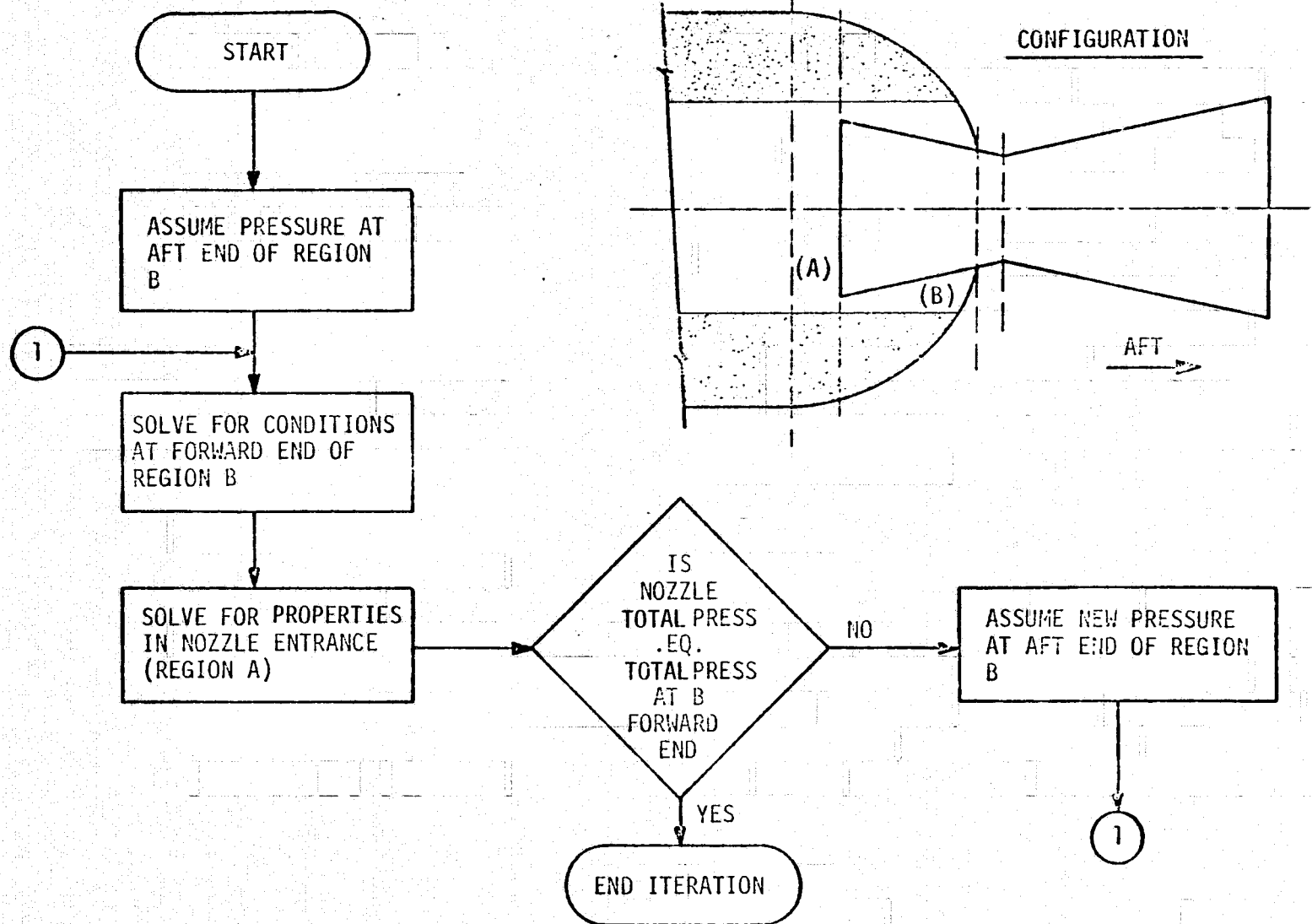


FIGURE 13-6 — MASTER ITERATION SCHEME FOR SUBMERGED NOZZLE GAS DYNAMIC SOLUTION

13.1.2 (Continued)

combustion products properties data are derived from available tabular input data as a function of pressure P1A. Subroutine XLIN is called to linearly interpolate values of total temperature (T0), specific heat ratio (GAMA), and molecular weight (AMW) for the assigned value of P1A. A new gas constant, R, is then calculated from the updated molecular weight, AMW.

Preliminary calculations are performed beginning with Statement 110 and continuing to Statement 120. The initial guess for the region B aft stagnation pressure is the region A entrance stagnation pressure, P2BT=P1AT. Next an estimate of density at the forward end of region B is made assuming stagnation conditions,

$$RH01B = P2BT / (12. * R * T0).$$

Burning rate calculations follow the density computations. The variable P is set to the reference pressure for burning rate calculation (in this case P=P2BT) and Subroutine RBSUB is called for the actual calculation of burning rate. The variable RB is returned from Subroutine RBSUB as the burning rate (NOTE: This general procedure is followed for all burning rate calculations in NSCE).

The mass generation term DWDOT is estimated for region B and the first guess for the velocity of gas at the forward end of B (U1B) is calculated from the steady state continuity equation. U1B carries a negative sign since its direction is toward the forward end of the motor being simulated. At this point the program logic chooses either a steady state solution (IOPT=2) or a non-steady state solution (IOPT=3). The module engineering and programming descriptions for IOPT=2 and 3 are given in the following paragraphs.

13.1.2.1 Steady State Solution of Submerged Nozzle Gas Dynamics (IOPT=2)

The steady state simulation is performed between Statements 120 and 190. The basic equations for region A are given in Table 13-II; basic equations for region B are given in Table 13-III. These equations are iterated in the steady state simulation. Initial guesses for the required iteration parameters (UTMP, P1B, PTMP, and RHOTMP) are set up between the test

IF(IOPT.EQ.3)GO TO 190

and Statement 120, and the iteration loop for pressure in region B is initiated.

The region B iteration loop begins with Statement 120 and continue to Statement 140. First the burning rates at the forward and aft ends of region B are generated using subroutine RBSUB. Next combustion gas properties are updated for pressure PTMP using Subroutine XLIN if tabular data has been input to IBM. At Statement 130, the mass generated

13.1.2.1 (Continued)

in region B, DWDOT, is calculated as a function of the average burning rate, burning area (AANB), and propellant density (DELF). A new estimate of forward-end velocity is then calculated from the steady state continuity equation:

$$UTMP = -DWDOT / (12 * RHOTMP * A1B)$$

New estimates of the forward-end static temperature (TTMP) and pressure (PTMP) are computed using the energy and momentum equations:

$$TTMP = T0 - (GAMA - 1.0) * UTMP^2 / (2 * GNOT * R * GAMA)$$

$$PTMP = P2BT - 2 * DWDOT * UTMP / (GNOT * (A1B + A2B))$$

Next a new calculated value of density at the forward end of region B (RH01B) is computed from the perfect gas equation. The variable TEMP is then formed and the test for convergence is applied.

IF(ABS(TEMP)/RHOTMP.LE..0001)GO TO 140

In iterations in which the test for convergence fails, the value RHOTMP is updated, and the iteration counter ITP1B is incremented, and a test for the maximum number of iterations is performed.

IF(ITP1B.LE.21)GO TO 120

If ITP1B.LE.21, the loop continues. In the event an iteration fails to converge (ITP1B.GT.21), an error message is generated, indicators are set which terminate the case and the program control returns to the calling routine.

On passing the test for convergence, the program logic branches to statement 140 and sets the values for forward-end conditions in region B to their last iterated values. Steady state calculations for region A are initiated at statement number 150. First the values of burning rate at the forward and aft-ends of region A are computed and the mass generation term for region A (DWDOT) is calculated as a function of average burning rate, propellant burning surface area (AANA) and propellant density (DELF). Next the flowrate into the nozzle entrance plane (MDBETA) is calculated using a form of the continuity equation. Properties data for the combustion products are updated for pressure P1A using subroutine XLIN if required and Subroutine SUBSON is called to check for subsonic flow at the nozzle throat. In the event the nozzle is subsonic and therefore not choked (ICH0KE=0), calculations for nozzle entrance static pressure (PBETA), nozzle stagnation pressure (PON), nozzle entrance static temperature (TBETA), sonic velocity at the nozzle entrance (CBETA), and velocity in the nozzle entrance (UBETA) are performed in Subroutine SUBSON. Program logic branches to Statement 170. If the flow is choked (ICH0KE=1), the routine continues with determination of nozzle entrance Mach number as a function of entrance area ratio (AR) and specific heat ratio (GAMA) by calling Subroutine Mach (Statement 160). Computations

13.1.2.1 (Continued)

are then made for the nozzle entrance temperature (TBETA), sonic velocity (CBETA), and velocity (UBETA) using isotropic Mach number relationships. Next the static pressure in the nozzle entrance (PBETA) is calculated using the momentum equation and the nozzle stagnation pressure (PON) is calculated from isentropic Mach number relationships.

Statement 170 is the test for overall convergence in the motor aft-end section. If the pressures PON and P1BT are equal ($\pm 0.01\%$), the program logic branches to statement number 300. In the event P1BT is not equal to PON within tolerance, the iteration counter ITP1B is set to zero and a new estimate of the aft-end stagnation pressure (P2BT) is made using the modified slope method. After the new value of P2BT has been obtained, the iteration counter ITP2B is incremented and checked in order to test for more iterations than are allowed. If the number of iterations is not greater than maximum allowed, the iteration for region B parameters is reinitiated at Statement 120. In the event that the maximum number of iterations allowed has been reached, an error message is generated from format statement 1010. Indicators are set for termination of the case and the program control returns to the calling routine.

Upon overall convergence in the motor aft-end section, the parameters required by IBM for head-end pressure iterations and final output of data are set to the final computed values. Flowrate exiting region is defined by WDOT=MDBETA; static and total pressures at the nozzle entrance plane are defined by P=PBETA and PO=PON. The velocity and static temperature are defined by U=UBETA and T=TBETA; Mach number and port area are defined by AMACH=MBETA and AP=A2A. The mass generated in region A is set by DWDOT=DWDOTA; average burning rate in region A is defined by RB=RBZ(NI+1). Program control then returns to the calling routine.

13.1.2.2 Non-Steady State Solution of Submerged Nozzle Gas Dynamics (IOPT=3)

The non-steady state simulation is performed between Statements 190 and 300. Solution techniques for the non-steady state case are very similar to the steady state case with the non-steady continuity and momentum equations substituted for the steady state equations. The basic non-steady continuity and momentum equations for regions A and B are given in Table 13-IV. In the IBM non-steady gas dynamic simulation, the temperature variables are assumed to be independent of time ($dT/dt=0$); the same assumption is employed here.

Initial guesses for the required iteration parameters for region B are made between Statements 190 and 200. These iteration parameters are:

C1B - Sonic velocity at forward end of region B

M1B - Mach number at forward end of region B

13.1.2.2 (Continued)

T1BG - Static temperature at forward end of region B
 P1BG - Static pressure at forward end of region B
 VGAZ - Volume of gas in region A for previous time step
 VGBZ - Volume of gas in region B for previous time step
 VGA - Volume of gas in region A for this time step
 VGB - Volume of gas in region B for this time step
 PMAZ - Mean pressure in region A for previous time step
 PMBZ - Mean pressure in region B for previous time step

The region B iteration loop begins with Statement 200 and continues to the calculation of the region B forward-end total pressure P1BT after Statement 270. First the burning rates at the forward and aft ends of region B are generated using Subroutine RBSUB. Next, the mass generated in region B (DWDOT) is calculated from the average burning rate, the propellant burning surface area (AANB), and the propellant density (DELF). The mean pressure (PMB) over region B is calculated from the current guessed value of static pressure at the forward end of region B (P1BG) and total region B aft-end pressure (P1BT), and the previous value of mean pressure (PMBZ) is initialized if the computation time (TIME) is zero. Mean temperature over region B (TMB) is defined by the arithmetic average of the current guessed value of temperature at the forward end of region B (T1BG) and the total temperature (T0). Next the mass storage term (DWDOT) for the continuity equation is calculated and the flowrate out of region B (MD1B) is computed. The velocity at the forward end of region B (U1B) is calculated and the mean value of velocity in B (UBM) is defined).

The non-steady momentum equation is solved for the region B forward static pressure (P1B) by first calculating the momentum storage terms (TERM1 and TERM2) and then computing P1B from

$$P1B = P2BT + (2. / (A1B + A2B)) * (TERM1 + TERM2 + U1B * MD1B / GNOT).$$

The nozzle stagnation pressure PON is calculated and a test for convergence of the region B forward total pressure is then performed. If the test fails, a new value of P1BG is estimated using the modified slope method and a test for maximum number of iterations is performed on ITP1B. If ITP1B.LE.21, the program logic branches back to the beginning of the iteration loop at Statement 200. In the event of failure to converge (ITP1B.GT.21), an error message is generated from format 1020, indicators are set to terminate the case, and program control returns to the calling routine.

Upon convergence of the forward static pressure, the program logic branches to Statement 230 where the forward-end velocity for region B (U1B) is updated and the combustion properties are interpolated for pressure P1B

13.1.2.2 (Continued)

using Subroutine XLIN for cases in which tabular properties data were input (NCSTR.GT.0). At Statement 240, the static pressure T1B is calculated from

$$T1B = T0 - (GAMA - 1.) * U1B^2 / (2. * GNOT * GAMA * R).$$

Next a check for temperature convergence is performed. If the temperature has not converged within the specified tolerance, a new estimate of temperature T1BG using the modified slope method of projection and the pressure iteration counter ITP1B is reinitialized to zero. A test is then performed to determine if the maximum number of temperature iterations has been exceeded. If the value of ITT1B is less than or equal to the maximum value, the program logic reinitializes the pressure iteration loop at Statement 200. In the event the temperature iteration counter is exceeded, an error message is generated, indicators are set to terminate the case, and program control returns to the calling routine.

Upon converging on temperature, the program logic branches to Statement 270 where the Mach number (M1B), density (RH01B), and total pressure (P1BT) at the forward end of region B are calculated. Next the combustion properties T0, GAMA, and AMW are derived for pressure equal to the static pressure at the forward end of region A (P1A) for case in which tabular data are available (NCSTR.GT.0). At Statement 280, the mean pressure over region A is computed and the previous time step value is set if computational time is zero. Next, the burning rates at the forward and aft ends of region A are generated using Subroutine RBSUB and the mass generation term (DWDOT) for region A is formed as the product of the average burning, the propellant burning surface area (AANA) and the propellant density (DELF). The mean temperature for region A is defined by the mass-weighted mean temperatures of the entering gases

$$TMA = (T1A * WDOT + T0 * DWDOT + T1B * MD1B) / (WDOT + DWDOT + MD1B)$$

and the mass flowrate across the nozzle entrance plane is calculated from the continuity equation

$$MDBETA = WDOT + DWDOT + MD1B - VGA * (PMA - PMAZ) / (12. * R * TMA * DELT) - PMA * (VGA - VGAZ) / (12. * R * TMA * DELT).$$

Subroutine SUBSON is called to check for subsonic flow at the nozzle throat. In the event the nozzle is subsonic and therefore not choked (ICH0KE=0), calculations for nozzle entrance static pressure (PBETA), nozzle stagnation pressure (PON), nozzle entrance static temperature (TBETA), sonic velocity at the nozzle entrance, (CBETA), and velocity in the nozzle entrance (UBETA) are performed in Subroutine SUBSON and program logic branches to Statement 285.

Sonic flow in the nozzle throat (ICH0KE=1) causes the routine to continue with the determination of nozzle entrance Mach number as a function of

13.1.2.2 (Continued)

entrance area ratio (AR) and specific heat ratio (GAMA) by calling Subroutine MACH. Computations are then made for TBETA, CBETA, UBETA and mean velocity over region A (UMA) and the static pressure in the nozzle entrance is calculated by

$$\begin{aligned} PBETA = & 1./AENT*(PIA*A1A-PIB*A1B+(PIA-PIB) \\ & *(A2A-A1A)/2.0-VGA*UMA*(PMA-PMAS) \\ & /(12.*GNOT*R*TMA*DELT)-PMA*UMA \\ & *(VGA-VGAZ)/(12.*GNOT*R*TMA*DELT) \\ & -UBETA*MDBETA/GNOT+U1A*WDOT/GNOT)+U1B*MD1B/GNOT \end{aligned}$$

Next nozzle stagnation pressure (PON) is computed and a test for overall convergence in the motor aft-end section is made at Statement 285. If the total pressures PIBT and PON have failed to converge, the iteration counters for region B static temperature and pressure are set to zero. A new guess at region B aft-end total pressure is then made using the modified slope method and a check is executed to determine whether the maximum number of iterations on region B aft-end pressure has been exceeded. In cases where the maximum number of iterations has not been exceeded (ITP2B.LE.21), the program logic branches to the beginning of the region B iteration loop at Statement 200. If a case fails to converge in the number of iterations allowed, an error message is generated from format 1040, indicators are set to terminate the case and program control is returned to the calling routine.

Upon successful convergence of the total pressures PIBT and PON, the program logic branches to Statement 300 where the parameters required by IBM for head-end pressure iterations and final output of data are set to required values. A complete description of the parameters set at Statement 300 is found in paragraph 13.1.2.1.

13.2 SUBROUTINE MACH (ARINPT, GAMA, M) ENGINEERING DESCRIPTION

Subroutine MACH calculates the subsonic Mach number (M) from an input area ratio (ARINPT) and specific heat ratio (GAMA). MACH is called from the subsonic analysis routine, the steady state analysis, and the non-steady state analysis routines of the NSCE Module. The statement function

$$AR(AG1) = (1./AG1)*(((2./(GAMA+1.))*(1.+((GAMA-1.)/2.))* (AG1**2)))**((GAMA+1.)/(2.*(GAMA-1.)))$$

is used to calculate the area ratio (AR) from the argument AG1. This expression for AR is derived from the isentropic Mach number versus area ratio relation in gas dynamic theory. The area ratio versus Mach number relation is broken into three regions for speed of computation:

ARINPT.GE.2.4
ARINPT.GE.1.1 (AND .LT.24)
ARINPT.GE.1.0 (AND .LT.1.1)

13.2 (Continued)

If the input area ratio does not fall into one of these regions, an error message is generated from format Statement 1 and the run terminates.

An initial bracketing guess of Mach number for input area ratios greater than or equal to 2.4 is made between Statements 10 and 20, and values for the calculated area ratios AR1 and AR2 are computed for these bracketing Mach numbers. Statement 23 is the test for convergence which forces the calculated area ratio AR2 to be equal to the input area ratio ARINPT within a tolerance of 0.0001. Upon convergence within the required tolerance, program logic branches to Statement 50 where the output variable M is set to the final computed value of M2 and the subroutine is exited. If the iteration fails to converge within the requirements, a new guess for M2 is made between Statements 23 and 25 using the modified slope method, and a test is performed to determine whether the iteration counter ITER has exceeded the maximum allowable value of 15. If ITER.LE.15, the program branches to Statement 20 for a new iteration. If ITER.GT.15, an error message is generated from Statement 25 and the run terminates.

Bracketing Mach number estimates for area ratios less than 2.4 and greater than or equal to 1.1 are made between Statements 30 and 40. The computations then enter the iteration loop at Statement 23. The bracketing Mach number estimates for ARINPT less than 1.1 and greater than or equal to 1.0 are made between Statements 40 and 50. The computations then enter the iteration loop at Statement 23.

13.3 SUBROUTINE SUBSON (ICHOKE, IOPT) ENGINEERING DESCRIPTION

Subroutine SUBSON tests for sonic flow at the nozzle throat and performs iterative computations to obtain the nozzle entrance Mach number (MBETA), static temperature (TBETA), sonic velocity (CBETA), velocity (UBETA), static pressure (PBETA) and total pressure (PON) for cases in which the flow is subsonic in the nozzle throat. One basic assumption implicit in the SUBSON analysis is that the nozzle stagnation pressure is closely approximated by the total pressure in the forward end of region A. SUBSON is called from both the steady and non-steady analysis routines of the NSCE Module. The initial part of the subroutine sizes and locates the Real and Common variables used in the subroutine.

The variable ICHOKE indicates subsonic (ICHOKE=0) or sonic flow (ICHOKE=1) at the nozzle throat. ICHOKE and the iteration loop counter ITER are initialized to zero before Statement 5. The multiplier variable STO is used to zero out the non-steady terms in the momentum equation when a steady state case is executed. STO is initialized to IOPT-2 before Statement 5. A test is then performed to check for vacuum ambient conditions (PA=0.0) at the nozzle exit plane. If vacuum conditions exist, the nozzle is choked and ICHOKE is set to 1 at Statement 5. Program control then returns to the calling routine.

D256-10020-1

13.3 (Continued)

For cases in which ambient pressure greater than zero (PA.GT.0.0), a first estimate of PON is made at Statement 10 and the critical pressure ratio for the nozzle throat (PRCRIT) is calculated as a function of specific heat ratio (GAMA). A test is made at Statement 15 for PA/PON less than or equal to PRCRIT. If (PA/PON.LE.PRCRIT), the nozzle is choked and the program logic branches to Statement 5. At Statement 5, ICHOKE is set to one and control returns to the calling routine.

In cases for which PA/PON is greater than PRCRIT, a calculation of Mach number at the nozzle exit (MEX) is made by

$$MEX = \text{SORT}((2. (GAMA-1.)) * ((PON/PA) ** ((GAMA-1.)/GAMA) - 1.))$$

If MEX is greater than or equal to 1.0, the program logic branches to Statement 5 and exits to the calling routine.

For subsonic exit cases (MEX.LT.1.0), the exit area ratio AEXAS is computed and the entrance area ratio AENTAS is calculated. A test is then performed on AENTAS to check for sonic flow in the throat and subsonic flow at the exit, i.e., the case of a normal shock in the nozzle. If AENTAS is less than 1.0, the throat is choked and program logic branches to Statement 5 to exit the subroutine.

For cases in which AENTAS is greater than or equal to 1.0, the flow at the throat is subsonic. The Mach number at the nozzle entrance (MBETA) is calculated in Subroutine MACH as a function of entrance area ratio (AENTAS) and specific heat ratio (GAMA). Next calculations for TBETA, CBETA, UBETA, UMA, PBETA and PONC (nozzle stagnation pressure) are made using isentropic Mach number relationships and the momentum equation. A check for pressure convergence is then performed on PON and PONC. If the pressure has converged within a tolerance of 0.001, program control returns to the calling routine. If the pressures have not converged, a new estimate of PON is made and a check is made to determine whether the maximum number of iterations has been exceeded. If the iteration counter ITER is less than 25, a new iteration is made by branching to Statement 15. If the maximum iterations have been exceeded, an error message is generated and program control returns to the calling routine.

13.4 NOZZLE SUBMERGENCE AND CONTOUR EFFECTS MODULE LIMITATIONS

The limitations on the use of NSCE are as follows:

- (1) The submerged nozzle entrance plane must be perpendicular to the motor centerline and centered in the port.
- (2) Region A must be defined large enough to allow stagnation of the reversed flow from region B.

D256-10020-1

SECTION 14

14.0 REFERENCES

1. NASA SP-273, "Computer Program for Calculation of Complex Chemical Equilibrium Compositions, Rocket Performance, Incident and Reflected Shocks, and Chapman-Jouquet Detonations", Sanford Gordon and Bonnie J. McBride, NASA Lewis Research Center, 1971.
2. AFRPL-TR-66-297, "[U] Impulse Scaling Prediction", Palmer W. Smith, George A. Stetz and Dr. Lawrence J. Delaney, Air Force Rocket Propulsion Laboratory, November 1966.
3. AFRPL-TR-71-7, "Solid Propellant Impulse Scaling Prediction Techniques", February 1971.
4. "Axisymmetric Two-Phase Perfect Gas Performance Program", C. R. Kliegel and G. R. Hickerson, TRW, April 1967.
5. Aerospace Corporation Computer Program, "IDNOZL", B. R. Kubert, November 13, 1964.
6. Boeing Document D2-125286-1, "Solid Propellant Rocket Motor Internal Ballistics Computer Program - Program Manual", February 28, 1967.
7. NASA Report, SP-8076, "Solid Propellant Grain Design and Internal Ballistics", March 1972.

APPENDIX

SOLID PROPELLANT ROCKET MOTOR
INTERNAL BALLISTICS COMPUTER PROGRAM
PROGRAM MANUAL REVISION

1.0 INTRODUCTION

This Appendix contains the revised pages and a list of the pages that should be deleted from Boeing Document D2-125286-1, "Solid Propellant Rocket Motor Internal Ballistic Computer Program - Program Manual", in order to make it applicable to the Internal Ballistics Module used in SRB-II. The revised pages should replace the pages with the same page number currently in the document. The deleted pages contain descriptions of parts of the computer program that are not used in the Internal Ballistics Module. Table A-1 contains a list of the pages to be deleted. Table A-2 contains a list of the pages to be revised.

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE A-1: PAGES TO BE DELETED

187 - 189

193 - 206

TABLE A-2: PAGES TO BE REVISED

x	77
xx	83
xx+	85
5-6	92-93
8	101
11-12	103-104
14-16	150
21-22	152-162
31	164-175
37	178-179
39	181-186
41	190-192
46	207-253
52	

[THESE PAGES TO BE REVISED ARE ATTACHED TO THIS DOCUMENT]

D256-10020-1

9-16-117

APPENDIX

NOT TO BE MAINTAINED

THE **BOEING** COMPANY

CODE IDENT. NO. 81205

ORIGINAL PAGE IS
OF POOR QUALITY

DOCUMENT NO. D2-125286-1 AS 2589

TITLE: Solid Propellant Rocket Motor Internal
Ballistics Computer Program - Program Manual

MODEL _____ CONTRACT NO. DA-01-021-AMC-15557(Z)

ISSUE NO. <u>1</u>	ISSUED TO <u>Charlie Carter</u>
--------------------	---------------------------------

ISSUED
JUN 20 1974

SEE DISTRIBUTION LIMITATIONS PAGE

PREPARED BY K.L. Johnson, F.M. Knox, G.D. Poe
K.L. Johnson, F.M. Knox, G.D. Poe

SUPERVISED BY H. J. Kuhls, Wm. Ward
H. J. Kuhls, Wm. Ward

APPROVED BY J. A. Brousseau
J. A. Brousseau

APPROVED BY J. S. Lesko 2/28/67
J. S. Lesko

APPROVED BY _____

BOEING

NO. D2-125286-1

SH. 1

LIMITATIONS

This document is controlled by Flight Technology Propulsion, 2-5711

All revisions to this document shall be approved by the above noted organization prior to release.

DDC AVAILABILITY NOTICE

- ☒ Qualified requesters may obtain copies of this document from DDC.
- ☐ Foreign announcement and dissemination of this report by DDC is not authorized.
- ☐ U. S. Government agencies may obtain copies of this document directly from DDC. Other qualified DDC users shall request through The Boeing Company, Seattle, Wn.
- ☐ U. S. military agencies may obtain copies of this document directly from DDC. Other users shall request through The Boeing Company, Seattle, Wn.
- ☐ All distribution of this document is controlled. Qualified DDC users shall request through The Boeing Company, Seattle, Wn.

PROPRIETARY NOTES

ACTIVE SHEET RECORD

SHEET NUMBER	REV LTR	ADDED SHEETS				SHEET NUMBER	REV LTR	ADDED SHEETS			
		SHEET NUMBER	REV LTR	SHEET NUMBER	REV LTR			SHEET NUMBER	REV LTR	SHEET NUMBER	REV LTR
I						23					
II						24					
III	A					25					
IV	A					26					
V	A					27					
VI						28					
VII	A					29					
VIII						30					
IX						31					
X						32	A				
XI						33	A				
XII						34					
XIII						35					
XIV						36					
XV						37	A				
XVI						38					
XVII						39	A				
XVIII						40					
XIX						41					
XX						42					
XXI						43					
1						44					
2						45					
3						46					
4						47					
5						48					
6						49					
7						50					
8						51					
9						52	A				
10						53					
11	A					54					
12						55					
13						56					
14	A					57					
15						58					
16						59					
17						60	A	60.1	A		
18						61					
19						62					
20						63					
21						64					
22						65					

ACTIVE SHEET RECORD

SHEET NUMBER	REV LTR	ADDED SHEETS				SHEET NUMBER	REV LTR	ADDED SHEETS			
		SHEET NUMBER	REV LTR	SHEET NUMBER	REV LTR			SHEET NUMBER	REV LTR	SHEET NUMBER	REV LTR
66						109					
67						110					
68						111					
69						112					
70						113					
71						114					
72						115					
73		73.1	A			116					
74	A					117					
75	A					118	A				
76						119					
77						120					
78						121					
79						122					
80						123					
81						124					
82						125					
83						126					
84						127					
85						128					
86						129					
87						130					
88						131					
89						132					
90						133					
91						134					
92						135					
93						136					
94						137					
95						138					
96						139					
97						140					
98						141					
99						142					
100						143					
101						144					
102						145					
103						146					
104						147					
105						148					
106						149					
107						150					
108						151					

ACTIVE SHEET RECORD

SHEET NUMBER	REV LTR	ADDED SHEETS				SHEET NUMBER	REV LTR	ADDED SHEETS			
		SHEET NUMBER	REV LTR	SHEET NUMBER	REV LTR			SHEET NUMBER	REV LTR	SHEET NUMBER	REV LTR
152						195					
153						196	A				
154						197	A				
155						198					
156						199					
157						200					
158						201					
159						202					
160						203					
161						204					
162						205					
163						206					
164						207					
165						208					
166						209					
167						210					
168						211					
169						212					
170						213					
171						214					
172	A					215					
173						216					
174						217					
175						218					
176						219					
177						220					
178						221					
179						222					
180						223					
181						224	A	224.1	A		
182						225					
183						226					
184						227					
185						228					
186						229					
187						230					
188						231					
189	A					232					
190	A					233					
191	A					234					
192						235					
193	A					236					
194						237					

ACTIVE SHEET RECORD

SHEET NUMBER	REV LTR	ADDED SHEETS				SHEET NUMBER	REV LTR	ADDED SHEETS			
		SHEET NUMBER	REV LTR	SHEET NUMBER	REV LTR			SHEET NUMBER	REV LTR	SHEET NUMBER	REV LTR
238											
239											
240											
241											
242											
243											
244											
245											
246											
247											
248											
249											
250											
251											
252											
253											
254											
255											
256											

REVISIONS			
LTR	DESCRIPTION	DATE	APPROVED
A	<p>Revised: 11, iv, v, vii, 11, 14, 32, 33, 37, 39, 52, 60, 74, 75, 118, 172, 189, 190, 191, 193, 195, 196, 224</p> <p>Added: 60.1, 73.1, 224.1</p>	8/7/67	<p><i>[Signature]</i></p> <p>9/18/67</p> <p><i>[Signature]</i></p> <p>9-27-67</p>

USE FOR TYPEWRITTEN MATERIAL ONLY

PREFACE

This document is volume one (D2-125286-1) of three volumes. It describes a FORTRAN IV digital computer program developed for analysis of solid rocket motor internal ballistics. Volume one, "Program Manual," explains the theory and describes the mathematical model, program capabilities and information necessary for program maintenance and revision. Volume two, "User's Guide," (D2-125286-2) describes program options, preparation of program input data and program output. Volume three, "Sample Case Results and Program Listings," (D2-125286-3) contains the sample case results and complete program listings of the computer program. The document is divided into three volumes for handling convenience. Section numbering is continuous through the three volumes. A complete table of contents appears in each volume.

CONTENTS

	<u>PAGE</u>
VOLUME 1 (D2-125286-1)	
Preface	viii
Abstract	xii
Key Words	xiii
Acknowledgements	xiv
Definition of Terms	xv
List of Illustrations	xvii
1.0 Introduction	1
1.1 General Information	1
1.2 History of Program Development	3
1.3 References	4
2.0 Program Capabilities and Limitations	5
2.1 Program Capabilities	5
2.1.1 Program Output	7
2.1.2 Program Assumptions	7
2.2 Limitations	8
3.0 Method of Solution	11
4.0 Gas Dynamic Equation Development	16
4.1 Gas Dynamics for Incremental Control Volumes	16
4.1.1 Steady Flow Gas Dynamics	19
4.1.2 Non-Steady Flow Gas Dynamics	23
4.1.3 Non-Steady Flow Gas Dynamics with Acceleration	33
4.2 Complete Motor Gas Dynamics	36
4.2.1 Fore-Head Pressure Convergence	36
4.2.2 Nozzle Gas Dynamics	39
4.3 Propellant Characteristics and Burning Rate Model	41
4.3.1 Propellant Gas Properties	41
4.3.2 Propellant Burning Rate Model	42
4.3.3 Anisotropic Propellant Burning	44
5.0 Geometrical Definition of Propellant Grain	56
5.1 Grain Cross Section Geometry	56

	<u>PAGE</u>
5.1.1 General Forked Wagon Wheel	56
5.1.2 Slotted-Cone	67
5.2 Grain Longitudinal Geometry	77
5.2.1 Head-End with Web, Fore-Head Section	77
5.2.2 Cylindrical Section	92
5.2.3 Straight-Through Grain, Motor End Sections	94
5.3 Moments of Inertia and CG Location	106
5.3.1 Roll Moment of Inertia	107
5.3.2 Pitch Moment of Inertia	108
5.3.3 CG of Sections	110
6.0 Detailed Programming Information	150
6.1 Subroutine Description	151
6.1.1 Subroutine Descriptions	151
6.1.2 Subroutine Linkage Table	161
6.2 Flow Charts	171
6.3 COMMON Allocation	188
6.4 Deleted	206
6.5 Nomenclature	207
7.0 Results	254
 VOLUME 2 (D2-125286-2)	
Preface	vi
List of Illustrations	xvii
8.0 User's Guide	1
8.1 Description of Input Data	2
8.1.1 Grain Geometry	2
8.1.2 Propellant Properties	8
8.1.3 Nozzle Configuration	11
8.1.4 Internal Gas Dynamics	12
8.2 Preparation of Input Data	14
8.3 Output Description	47
8.3.1 Input Data Printout	47
8.3.2 Initial Geometry Printout	47

	<u>PAGE</u>
8.3.3 Geometry Table Printout	47
8.3.4 Inert Sliver Summary Printout	47
8.3.5 Internal Ballistic Solution Printout	48
8.3.6 Expanded Increment Dividing Plane Printout	51
8.3.7 Burnout Printout	52
8.4 Sample Cases	54
8.4.1 Base Case One and Variations Description	54
8.4.2 Base Case Two and Variations Description	55
8.5 Operating Information	57
VOLUME 3 (D2-125286-3)	
Preface	xii
List of Illustrations	xvi
9.0 Sample Case Results and Program Listings	1
9.1 Sample Case Results	1
9.2 Program Listings	313

USE FOR TYPEWRITTEN MATERIAL ONLY

ABSTRACT

This report describes a FORTRAN IV digital computer program developed to calculate internal ballistic performance of solid propellant rocket motors with high burning rates, short burning durations, and high vehicle accelerations. Forked wagon wheel, conventional wagon wheel, standard star, slotted-cone, and circular port monolithic and segmented grain designs may be considered. Accurate description of an inert sliver in the cylindrical section is allowed for all but the forked wagon wheel grain design. The effects of anisotropic burning of the propellant may be considered. The storage of mass and momentum (capacitance effects) and vehicle acceleration are included in the internal gas dynamic equations. Ignition transients may be calculated. Tabular input of the motor grain description is available for special grain configurations that cannot be described by the program geometry constants. Motor performance parameters such as delivered and vacuum thrust and total impulse, fore-head and aft-head total pressure, nozzle discharge flow, fore-head and aft-head total pressure integrals, pitch and roll moments of inertia, center of gravity locations, burn surface area, and weight of propellant remaining are printed for each time interval.

This report is divided into three volumes. Volume one is the Program Manual, volume two is the User's Guide, and volume three is the Sample Case Results and Program Listings.

KEY WORDS

The following Key Words identify the major program capabilities:

Internal Ballistics
Solid Propellant Rocket Motor
Monolithic Grain
Segmented Grain
One-Dimensional Gas Dynamics
Steady Flow Gas Dynamics
Non-Steady Flow Gas Dynamics
Isotropic Propellant Burning
Anisotropic Propellant Burning
Vehicle Acceleration
Ignition Transient Interval
Web-Time Interval
Tail-Off Interval
Fore-Head Section
Aft-Head Section
Cylindrical Section
Center of Gravity
Moment of Inertia
Grain Geometry
Nozzle

ACKNOWLEDGEMENTS

Acknowledgement is given to the Thiokol Chemical Corporation, Redstone Division, for the original SCAT machine language program from which this program was developed (References 1 and 2), and to the Thiokol Chemical Corporation, Wasatch Division, for the documentation of the grain geometry calculations from which the description of the mathematical model was developed (Reference 3).

DEFINITION OF TERMS

The following general definitions apply throughout this document:

Internal Ballistics: Analysis of the burning characteristics and progression of the propellant surface, dynamics of the gas flow, and the gas generation in the interior of solid propellant rocket motors.

Performance Characteristics: Parameters that specify motor performance, e.g., thrust vs time, maximum chamber pressure, specific impulse, and burn time.

Gas Dynamics: Study of the generation and flow of combustion products along the propellant grain and through the nozzle.

Steady Flow Gas Dynamics: Mass, energy, and momentum within a control volume are constant with time.

Non-Steady Flow Gas Dynamics: Mass, energy, and momentum within a control volume are not constant with time.

Grain Design: The cross sectional grain configuration of the propellant.

Monolithic Grain: The propellant grain is one single piece.

Segmented Grain: The propellant grain is divided up into a number of longitudinal segments.

Slots: The region between the segments which does not contain propellant.

Web: The minimum distance between the grain surface and the case wall.

Core: The region occupied by the combustion gases.

Reference Planes: The stations in the cylindrical section of a motor where the grain design is specified.

Increment Dividing Planes: The stations in the cylindrical section of the motor where the solution of the internal gas dynamic equations is obtained.

Mass Addition Region: The region between increment dividing planes.

Ignition Transient Interval: The time required to obtain motor operating pressure.

Web-Time Interval: The time required to burn through the web.

Tail-off Interval: The time interval after web burn through.

Isotropic Propellant Burning: Where the burning rate characteristics are independent of distance burned.

Anisotropic Propellant Burning: Where the burning rate characteristics are dependent on distance burned.

LIST OF ILLUSTRATIONS

<u>FIGURE NUMBER</u>	<u>TITLE</u>	<u>PAGE</u>
2.1	Grain Design Options	9
2.2	Typical Motor Configurations	10
3.1	Reference Plane and Increment Dividing Plane Identification	14
3.2	Macroscopic Flow Chart	15
4.1	Mathematical Model of Mass Addition Region Control Volume	50
4.2	Mathematical Model of Slot Between Grain Segments	51
4.3	Reference Plane and Increment Dividing Plane Identification	52
4.4	Anisotropic Propellant Burning Configuration Near Case Wall Before Web Burnout	53
4.5	Anisotropic Propellant Burning Configuration Near Case Wall and Inert Sliver Before Web Burnout	54
4.6	Anisotropic Propellant Burning Configuration Near Case Wall After Web Burnout	55
5.1	Grain Design Options	112
5.2	One-Half Fork of General Modified Wagon Wheel Configuration	113
5.3	Part of Calculated Constants for One-Half Fork of General Modified Wagon Wheel Confi- guration by PLNCNS Subroutine	114
5.4	Part of Calculated Constants for One-Half Fork of General Modified Wagon Wheel Configuration Produced by PLNCNS Subroutine	115
5.5	Sector Definition	116
5.6	Slotted-Cone	117

USE FOR TYPEWRITTEN MATERIAL ONLY

<u>FIGURE NUMBER</u>	<u>TITLE</u>	<u>PAGE</u>
5.7	Slotted-Cone Addition to Standard Star Showing Location of Fixed Geometry Points	118
5.8	Complete Slotted-Cone Burning Surface for τ Less than R_2	119
5.9	Slotted-Cone Burning Surface Addition for τ Greater than R_2 and Less than $T_{6\max}$	120
5.10	Slotted-Cone Burning Surface Addition for τ Greater than $T_{6\max}$ and Less than $T_{7\max}$	121
5.11	Slotted-Cone Burning Surface Addition for τ Greater than $T_{7\max}$ and Less than T'_{V7} with T'_{V7} Less than T'_{V6}	122
5.12	Slotted-Cone Burning Surface for τ Greater than T'_{V7} with T'_{V7} Less than T'_{V6}	123
5.13	Slotted-Cone Burning Surface for τ Greater than T'_{V6} with T'_{V6} Less than T'_{V7}	124
5.14	Head-End with Web, Motor Fore-Head	125
5.15	Sectors for Block No. 1	126
5.16	Head-End with Web Plane Definition	127
5.17	Head-End with Web, Block 1 Plane Definition	128
5.18	Plane for Block 1 Analysis	129
5.19	γ_1 for Subroutine GAMSUB	130
5.20	γ_2 for Subroutine GAM2S	131
5.21	P_0 for Subroutine POSUB	132
5.22	P_3 for Subroutine P3SUB	133
5.23	Planes for Block 1 Analysis	134
5.24	Planes Produced in Sectors 3A and 3B or in Sectors 11A and 11B	135
5.25	Area for Block No. 2A and 2B	136
5.26	A_{1g} for Subroutine AIGSUB	137

USE FOR TYPEWRITTEN MATERIAL ONLY

<u>FIGURE NUMBER</u>	<u>TITLE</u>	<u>PAGE</u>
5.27	Sectors for Block 2B of Fore-Head Section	138
5.28	Sector for Block 2B	139
5.29	Distribution of Volume In Radial Burning Section	140
5.30	Motor Case Longitudinal Constants	141
5.31	Straight Through Grain Configuration	142
5.32	Sector Definition	143
5.33	Zone Definition	144
5.34	Sector Parameters, Zones B, C, and Web	145
5.35	Circular Arc Sector 3 Zone A	146
5.36	Element of Incremental Area for Zone B	147
5.37	Head-End Section, Volume Elements for MOI and CG Calculations	148
5.38	Cylindrical Section Volume Elements for MOI and CG Calculations	149
7.1	HIBEX Forehead Chamber Pressure Trace	255
7.2	Influence of Internal Flow and Burning Rate Model	256
8.1	Program Model	62
8.2	Typical Motor Configurations	63
8.3	Straight Through Grain Configuration	64
8.4	Straight Through Grain Configuration Inputs	65
8.5	Head-End Web	66
8.6	Grain Configuration Options	67
8.7	Reference Plane and Increment Dividing Plane Identification	68
8.8	General Forked Wagon Wheel Configuration Inputs	69

USE FOR TYPEWRITTEN MATERIAL ONLY

ORIGINAL PAGE IS
OF POOR QUALITY

NUMBER D2-125206-1
REV LTR

THE **ROBINS** COMPANY

<u>FIGURE NUMBER</u>	<u>TITLE</u>	<u>PAGE</u>
8.9	Calculated Plane Constants	70
8.10	Calculated Plane Constants	71
8.11	LA Definition	72
8.12	Wagon Wheel Configuration Inputs	73
8.13	Star Grain Configuration Inputs	74
8.14	Slotted-Cone Grain Configuration Inputs	75
8.15	Circular Port Configuration Inputs	76
8.16	Nozzle Configuration Inputs	77
8.17	Motor Case Longitudinal Inputs and Constants	78
8.18	Base Case One	79
8.19	Base Case Two, Monolithic Grain	80
8.20	Base Case Two, Segmented Grain	81

LIST OF FLOW CHARTS - VOLUME 1

<u>FLOW CHART NUMBER</u>	<u>TITLE</u>	<u>PAGE</u>
1	MAIN Program	172
2	Subroutine MNCHN1	173
3	Subroutine MNCHN2	174
4	Compute Plane Constants	175
5	Subroutine AESUB	176
6	Subroutine MNCHN3	178
7	Subroutine SCI	179
8	Subroutine SCTOR1	180
9	Subroutine SZOK	181
10	Subroutine MNCHN4	182

SHEET

xx

ORIGINAL PAGE IS
OF POOR QUALITY

NUMBER D-1252-1-3
REV. 1/68

THE ~~SECRET~~ ~~SECRET~~

<u>FLOW CHART NUMBER</u>	<u>TITLE</u>	<u>PAGE</u>
11	Subroutine SEGSUB	183
12	Subroutine SETPH	185
13	Subroutine TISUB	186

USE FOR TYPEWRITTEN MATERIAL ONLY

1.0

INTRODUCTION

This report describes a FORTRAN IV digital computer program, developed by The Boeing Company for operation on an IBM 7094 computer with an IBSYS version 13 monitor system from an earlier Thiokol Chemical Corporation program, to perform dynamic analyses of solid propellant rocket motor internal ballistics. A complete description of the mathematical model, the engineering equation development, the method of solution, and detailed programming information is presented to explain program principles and theory and to facilitate program maintenance and revision.

This program was developed under contract from the Army Missile Command Propulsion Laboratory, Redstone Arsenal, Alabama, contract number DA-01-021-AMC-15557(Z) by the Engineering Digital Computing Organization 2-2640, with support from the Missile and Information Systems Division, Flight Technology Propulsion Organization 2-5711 of The Boeing Company.

1.1

General Information

The report is divided into three volumes. Volume I, the Program Manual, provides a technical explanation of the theory, mathematical model, program capabilities and information necessary for program maintenance and revision. Volume II, the User's Guide, describes program options, preparation of input data and program output. Volume III, the Sample Case Results and Program Listings, contains the complete program listings of the computer program. The following paragraphs describe briefly the sections of the report comprising each volume.

Volume I

The Program Capabilities and Limitations, Section 2.0, indicates the capability the program has to evaluate grain designs and internal ballistics and the program limitations that exist in these areas.

The Method of Solution, Section 3.0, describes the method the program uses to obtain the internal ballistic solutions and the organization of major program sections which divide the solution into logical blocks or core loads that reside in core at separate times.

The Gas Dynamic Equation Development, Section 4.0, presents the development of the equations for the non-steady flow gas dynamics for both segmented and monolithic motors, propellant description, and for an accelerating reference system. In general, this section describes modifications made to the original Thiokol Chemical Corporation program for conducting design studies of solid propellant configurations (Reference 1) to simulate the

Internal ballistics of high burning rate propellants with characteristically short burn durations. These modifications were made specifically to include the storage of mass and momentum in the gas dynamic equations, to consider the effects of very high vehicle acceleration on the internal ballistics, and to study anisotropic propellant burning. As a result of these modifications, ignition transients may be calculated.

The Geometry Equation Development, Section 5.0, explains in detail the setup and solution of the grain geometry equations which determine the perimeter length, cross sectional area, burn surface area, moments of inertia, and center of gravity location of the various grain options and longitudinal configurations. The equation development and figures presented in this section were obtained from References 1 and 3.

The Detailed Programming Information, Section 6.0, presents a brief description of all program subroutines, macroscopic program logic flow charts, a description of the computing system and program storage allocation, program diagnostic aids, and a list of the program nomenclature. Appropriate comments are placed throughout the program listings as a supplement to the subroutine flow charts to assist in program maintenance and revision.

The Results, Section 7.0, presents a comparison of a computer prediction with three full scale HIBEX motor firings. Dimensionless fore-head pressure traces are shown for the computer prediction and the motor firings.

Volume II

The User's Guide, Section 8.0, presents an explanation of the required program inputs, sample cases showing the available program options, a description of the output format, and the required control cards to permit effective program use and operation without knowledge of the program technical aspects. This section is arranged to be complete without reference to the program manual technical sections and may be used independent of the program manual.

Volume III

The Listings, Section 9.0, contains the sample case results and program listings of the computer program.

USE FOR TYPEWRITTEN MATERIAL ONLY

1.2

History of Program Development

In March 1960, work was initiated at the Thiokol Chemical Corporation, Redstone Division, under the auspices of Systems Analysis Laboratory, Army Rocket and Guided Missile Agency, Redstone Arsenal, Alabama, for the development of a solid propellant rocket motor design program (References 1 and 2). In 1962, The Boeing Company received a copy of this program, and in 1963 developed a segmented motor version. In 1964, the Saturn Branch of the Aerospace Division of Boeing at New Orleans converted the SCAT machine language program to FORTRAN II for operation on the IBM 7094 computer. In 1965, The Boeing Aerospace Division in Seattle, Washington, converted the FORTRAN II version to FORTRAN IV for operation on the SRU 1107 and made the modifications discussed in Section 1.1 for the HIBEX contract to add the transient capability. In May 1966, The Boeing Company proposed to the Army Missile Command to segment the existing SRU 1108 program, perform the necessary conversion for operation on the IBM 7094, and completely document the advanced program version. A contract was received in July 1966 from the Army Missile Command for a 6 month development effort to perform the required conversion.

USE FOR TYPEWRITTEN MATERIAL ONLY

1. Control No. U-A-61-28A, "Final Report Design Study of Solid Propellant Configurations," Thiokol Chemical Corporation, Redstone Division, July 1961.
2. Control No. U-A-62-18A, "Final Report, Design Study of Solid Propellant Configuration," Thiokol Chemical Corporation, Alpha Division, May 1962.
3. T263206, "Generalized Wagon Wheel Grain Design Study," Thiokol Chemical Corporation, Wasatch Division, June 1962.
4. D2-99598-1, "HIBEX Rocket Motor Performance, Design and Development Analysis (U)," Confidential, Aero-Space Division, The Boeing Company, March 1966.
5. D2-125060-2, "Small High Performance Interceptor Research (U)," Confidential, Aero-Space Division, The Boeing Company, December 1966.
6. Leipmann, H. W., and Roshko, A., Elements of Gas Dynamics, GALCIT Aeronautical Series, John Wiley and Sons, Inc., Third Printing, February 1960.
7. Shapiro, Ascher H., The Dynamics and Thermodynamics of Compressible Fluid Flow, Volume I, The Ronald Press Co., 1953.
8. Baumeister, Theodore, Mark's Mechanical Engineer's Handbook, Sixth Edition, McGraw-Hill Book Company, 1958.
9. Univac 1107 Fortran Programmers Guide, U-3540, August 1963.

USE FOR TYPEWRITTEN MATERIAL ONLY

2.0 PROGRAM CAPABILITIES AND LIMITATIONS

This computer program was developed to calculate solid propellant rocket motor internal ballistics. Because of its development from earlier grain design and ballistic performance programs, additional capabilities are present. Throughout the development effort, all prior program capabilities have been retained so that a general program exists with both grain design and internal ballistic evaluation capability.

2.1 Program Capabilities

The basic propellant grain design is the forked wagon wheel, however, the grain design equations are general so that the conventional wagon wheel, standard star and circular port as well as the slotted-cone may be described by variations in the input data. Figure 2.1 shows the five grain design options. Other more complicated grain designs may be evaluated by describing the perimeter length and burn area as a function of distance burned and input to the computer program as tabular data.

The propellant grain configuration may be either monolithic or segmented with up to 18 slots. The propellant case and port cavity may be either cylindrical or tapered. The fore-head section configuration may be a straight through grain or may contain a complete web. Figure 2.2 shows the various motor configuration options. The aft-head section configuration is a straight through grain.

The propellant characteristics are described by definitive properties of the combustion gases and a generalized burning rate equation. The propellant gas properties may be held constant or may be varied as a function of the static pressure in the port cavity. The burning rate model includes erosive burning and will allow either isotropic or anisotropic burning of the propellant surface.

Either steady or non-steady flow gas dynamics are available to obtain the internal ballistic solution. The steady flow gas dynamics solve the momentum and continuity equations without consideration of time dependent terms such that there is no storage of mass or momentum (no capacitance effects). The non-steady flow gas dynamics solve the momentum and continuity equations with respect to time so that the storage of mass and momentum is considered and the start transient and tail-off intervals may be determined.

The effects of vehicle acceleration on the internal ballistic solution may be considered. The acceleration term is included in the momentum equation.

The effect of a tapered inert sliver may be considered in the cylindrical section for all but the forked wagon wheel grain design.

USE FOR TYPEWRITTEN MATERIAL ONLY

2.1 Program Capabilities (Continued)

The following is a summary of the program capabilities:

1.0 Grain Design

- A. Circular Port
- B. Standard Star
- C. Slotted-Cone
- D. Conventional Wagon Wheel
- E. Forked Wagon Wheel
- F. Tables of Perimeter and Burn Area as a Function of Distance Burned can be Input

2.0 Motor Configuration

- A. Cylindrical Section
 - 1) Monolithic grain
 - 2) Segmented grain
 - 3) Tapered inert sliver
- B. Fore-head Section
 - 1) Straight through grain
 - 2) Complete web (head-end with web)
 - 3) Tabular data
- C. Aft-head Section
 - 1) Straight through grain
 - 2) Burning on aft face
 - 3) Tabular data

3.0 Propellant Characteristics

- A. Isotropic Propellant Burning
- B. Anisotropic Propellant Burning
- C. Erosive Burning
- D. Variable Gas Properties (function of pressure)

4.0 Internal Ballistics

- A. Steady Gas Flow
- B. Non-Steady Gas Flow
 - 1) Ignition transient interval
 - 2) Web-time interval
 - 3) Tail-off interval
- C. Vehicle Acceleration

2.1.1

Program Output

The program provides the following output:

1.0 Motor

- A. Delivered Thrust
- B. Vacuum Thrust
- C. Fore-head Total Pressure
- D. Nozzle Total Pressure
- E. Fore-head Total Pressure Time Integral
- F. Nozzle Total Pressure Time Integral
- G. Delivered Total Impulse
- H. Vacuum Total Impulse
- I. Nozzle Discharge Flow Rate
- J. Polar and Rectangular Moment of Inertia
- K. Center of Gravity

2.0 Propellant Characteristics

- A. Weight of Propellant Remaining
- B. Forward Tangent Plane Propellant Burning Rate
- C. Aft Tangent Plane Propellant Burning Rate
- D. Total Weight of Propellant Expended

3.0 Grain Geometry

- A. Cylindrical Section Burn Area
- B. Fore-head Section Burn Area
- C. Aft-head Section Burn Area
- D. Grain Segment Face Burn Area
- E. Total Motor Burn Area

4.0 Nozzle Characteristics.

- A. Throat Area
- B. Effective Expansion Ratio (flow separation accounted for)
- C. Pressure Ratio Across Nozzle
- D. Momentum Portion of Thrust Coefficient

2.1.2

Program Assumptions

The following assumptions were made to translate the physical system into the one-dimensional gas dynamic model:

1. Propellant burning during ignition and steady state operation occurs normal to the grain surface.
2. The burn rate in the fore-head and aft-head sections is assumed to be constant over the burning surface of the entire section.

2.1.2

Program Assumptions (Continued)

3. Mass addition occurs instantaneously with no velocity component along the motor's longitudinal axis ($dz/dt = 0$).
4. The products of combustion obey the perfect gas law.
5. The gas flow is one-dimensional and adiabatic.
6. For a given cross-section perpendicular to the longitudinal axis, the combustion temperature is constant.
7. The heat capacity of the combustion gases is constant.
8. The friction forces of the combustion gases in the port cavity are negligible.
9. The moments of inertia about the pitch and yaw axis are equal.

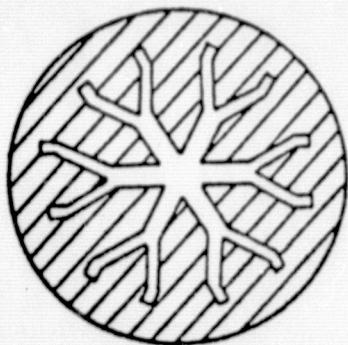
2.2

Limitations

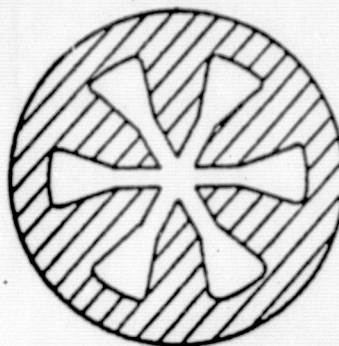
The program has the following limitations:

1. A maximum of 18 reference planes are allowed in the cylindrical section to describe the grain design.
2. A maximum of 100 increment dividing planes are allowed in the cylindrical section to define the mass addition regions. If this restriction is exceeded by defining a ΔZ too small, the case execution will be terminated and an appropriate comment will be printed.
3. A maximum of 18 slots are allowed for segmented motors.
4. The slotted-cone grain design is applicable only to the cylindrical section. Burn area tables must be input for the forward and aft domes when this grain design is used.
5. The inert sliver option is restricted to the cylindrical section and does not apply to either end section.
6. The effects of an accelerating reference system can be determined only for non-steady flow gas dynamics.

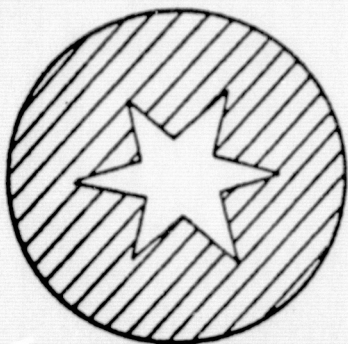
USE FOR TYPEWRITTEN MATERIAL ONLY



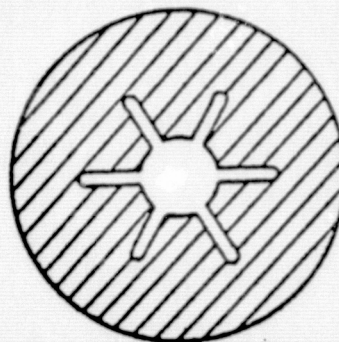
Forked Wagon Wheel



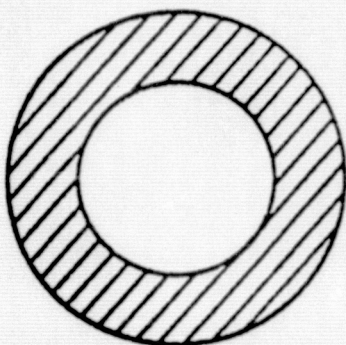
Conventional Wagon Wheel



Standard Star

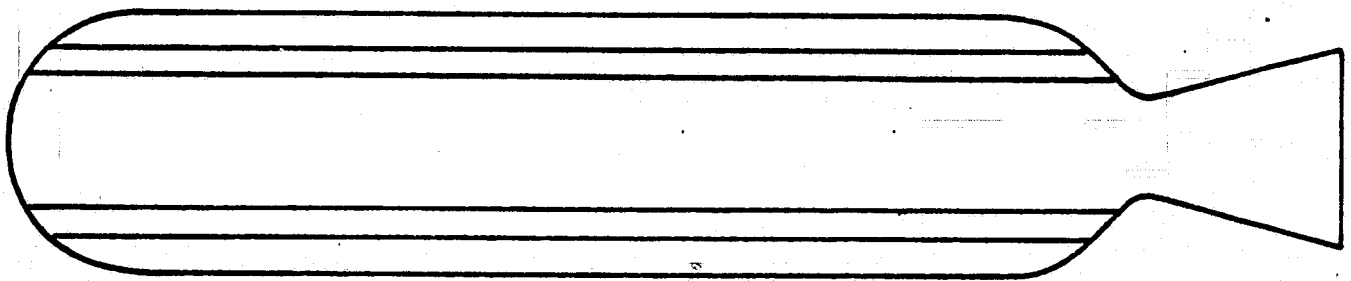


Slotted-Cone

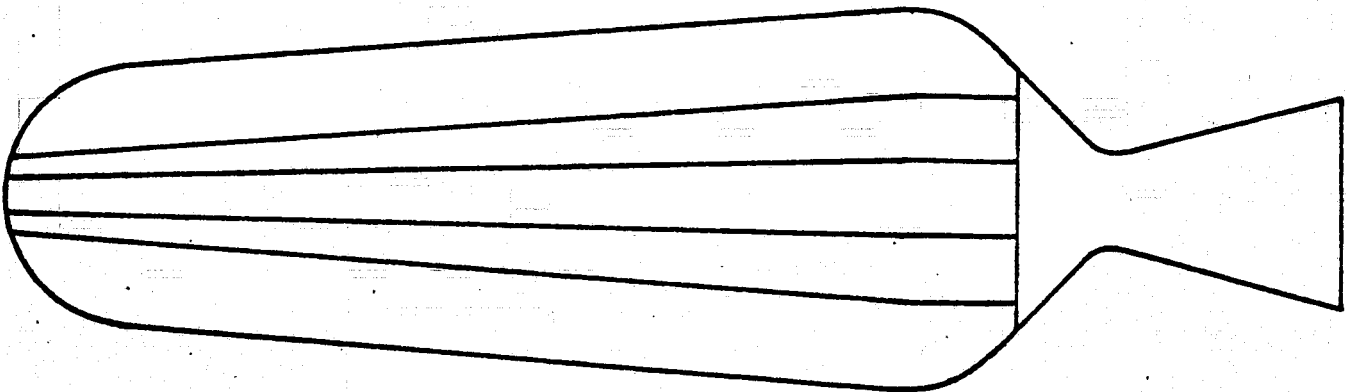


Circular Port

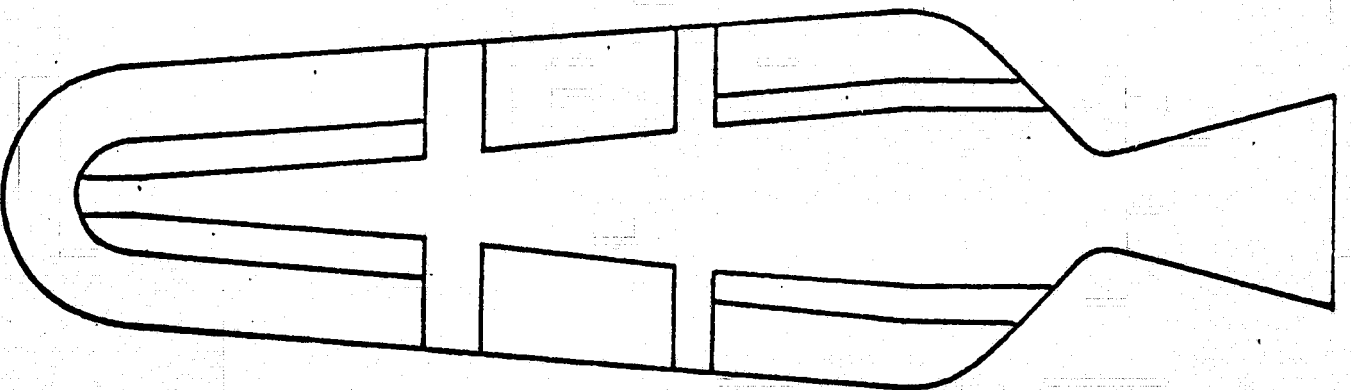
Figure 2.1. Grain Design Options



Monolithic Grain With Cylindrical Case



Monolithic Grain with Tapered Case
and Propellant Cutback in Aft-end



Segmented Grain with a Head-end with Web and Tapered Case

3.0

METHOD OF SOLUTION

The program calculations are based on the geometrical model shown in Figure 3.1. The motor configuration is divided into three sections: the head-end section (forward dome), the cylindrical section, and the aft-head section (aft dome) or nozzle. The grain geometry is described by input reference planes within the cylindrical section. The cylindrical section, which may contain either a monolithic or a segmented grain, is further divided into a number of increments or mass addition regions by the location of increment dividing planes, at each reference plane and at specified intervals (ΔZ) from each reference plane until either a segment slot interface or the next reference plane is passed. During the computer solution of the gas dynamics, port perimeter, port cross sectional area and moments of inertia are determined at each increment dividing plane by linear interpolation between adjacent reference planes. Mass addition is assumed to occur as a step process between two increment dividing planes.

The program method of solution is divided into four separate control routines linked together by a main control program and a common data region. Each routine is unique, but dependent on preceding routines for generated data. The flow chart shown in Figure 3.2 presents the macroscopic program order of solution with the separate routines linked together.

The first control routine contains the subroutines required to read the input data, initialize the data cells, compute the input reference plane constants, locate the increment dividing planes, check for input data errors, and print the program inputs and computed constants.

The second control routine contains the geometry subroutines required to compute the initial propellant cross section area and perimeter length for the cylindrical section reference planes, the aft-head and straight through grain fore-head sections burn area and initial propellant volume as a function of distance burned. The moments of inertia and centers of gravity for the aft-head and straight through grain fore-head sections and the radius of gyration for the cylindrical section are also calculated. The computed values for each section are stored in tables for use during the solution of the internal ballistics in the fourth control routine.

The third control routine contains the geometry subroutines required to compute the initial propellant volume, burn area, moments of inertia, and CG location tables of the head-end with

USE FOR TYPEWRITTEN MATERIAL ONLY

3.0

Method of Solution (continued)

web. The third routine is executed only if a head-end with web is required after the cylindrical section and aft-head section geometry calculations have been completed in the second routine.

After the grain geometry calculations are performed in the first, second, and third routines and the perimeter and area tables have been established, the internal ballistic solution is initiated in the fourth routine by determining the geometry values of each reference plane and each end section from a table look-up procedure in the geometry tables. For steady flow conditions, an initial estimate of the fore-head pressure is made, and the burning rate (which is assumed to be constant over the entire head-end section) is determined as described in Section 4.1.1. With this burn rate and the tabular value of the fore-head burn area, the instantaneous value of mass addition is determined for the fore-head section. The state and gas dynamic properties of the propellant at the forward tangent plane, the first increment dividing plane, are determined from the simultaneous solution of the momentum and continuity equations assuming perfect gas relationships.

The grain geometry at the first increment dividing plane is then determined and stored in temporary locations for future reference. The grain geometry of increment dividing plane two is determined. The increment section mass generation rate is determined from the perimeter lengths of increment dividing planes one and two, the increment length, and the burning rates at the upstream adjacent increment dividing planes. The propellant properties and mass flow at increment dividing plane two are then determined by a simultaneous solution of the momentum and continuity equations for either steady or non-steady flow conditions.

The above procedure is repeated for each cylindrical section mass addition region. Once the cylindrical section is complete, the same general calculations are performed for the aft-head section. The port cross-sectional area and perimeter length and burning rate are assumed to be constant in the aft-head section and identical to the values at the aft tangent plane.

The nozzle throat area is compared to the maximum value that will maintain subsonic flow in the port. If this maximum value is exceeded, the program prints an error comment and terminates the case. If this maximum value is not exceeded, the flow rate of propellant discharge through the nozzle is computed on the basis of isentropic flow. This flow rate is compared to the flow rate of propellant discharged from the grain. If these two values do not agree within .1 percent, the fore-head pressure is adjusted and the program returns to the fore-head and repeats until

convergence is attained. Once equilibrium is reached, additional ballistic properties are computed and the performance data is printed.

Following the performance printout, the thickness burned in each increment dividing plane and slot interface is then determined from the previous web thickness. A check is then made to see if burnout has occurred at any of the increment dividing planes. If burnout occurs, a comment is printed that the increment dividing plane has burned out. The progression of the slot interfaces for segmented motors is indicated by a printout of the increment dividing plane longitudinal location.

The time is then incremented by the computed time interval and the program returns to the fore-head to compute new equilibrium conditions and determine new values of the perimeter length and port cross section area at each increment dividing plane and burn area for each end section. This process is then repeated until the termination option is exceeded.

The general program solution of the internal ballistics outlined above is modified for non-steady flow conditions. When the start transient interval is computed, the fore-head pressure is defined by tabular input of the fore-head pressure as a function of time, or the burn rate coefficient is defined by tabular input of the burn rate coefficient as a function of distance burned. When the fore-head pressure is input for the start transient, the burn rate coefficient is varied to obtain convergence; and when the burn rate coefficient is input for the start transient, the fore-head pressure is varied, as for steady flow conditions, to obtain convergence. When the fore-head pressure is input for the start transient, an initial estimate of the burn rate coefficient is made by computing a first guess of the burn rate coefficient from the motor configuration parameters and the fore-head pressure variation. With this burn rate coefficient, the instantaneous value of mass addition and mass discharge is determined for the head-end section. The propellant gas properties for the first increment dividing plane are then determined from a simultaneous solution of the non-steady gas flow equations as above. The remainder of the ballistic solution is unchanged.

ORIGINAL PAGE IS
OF POOR QUALITY

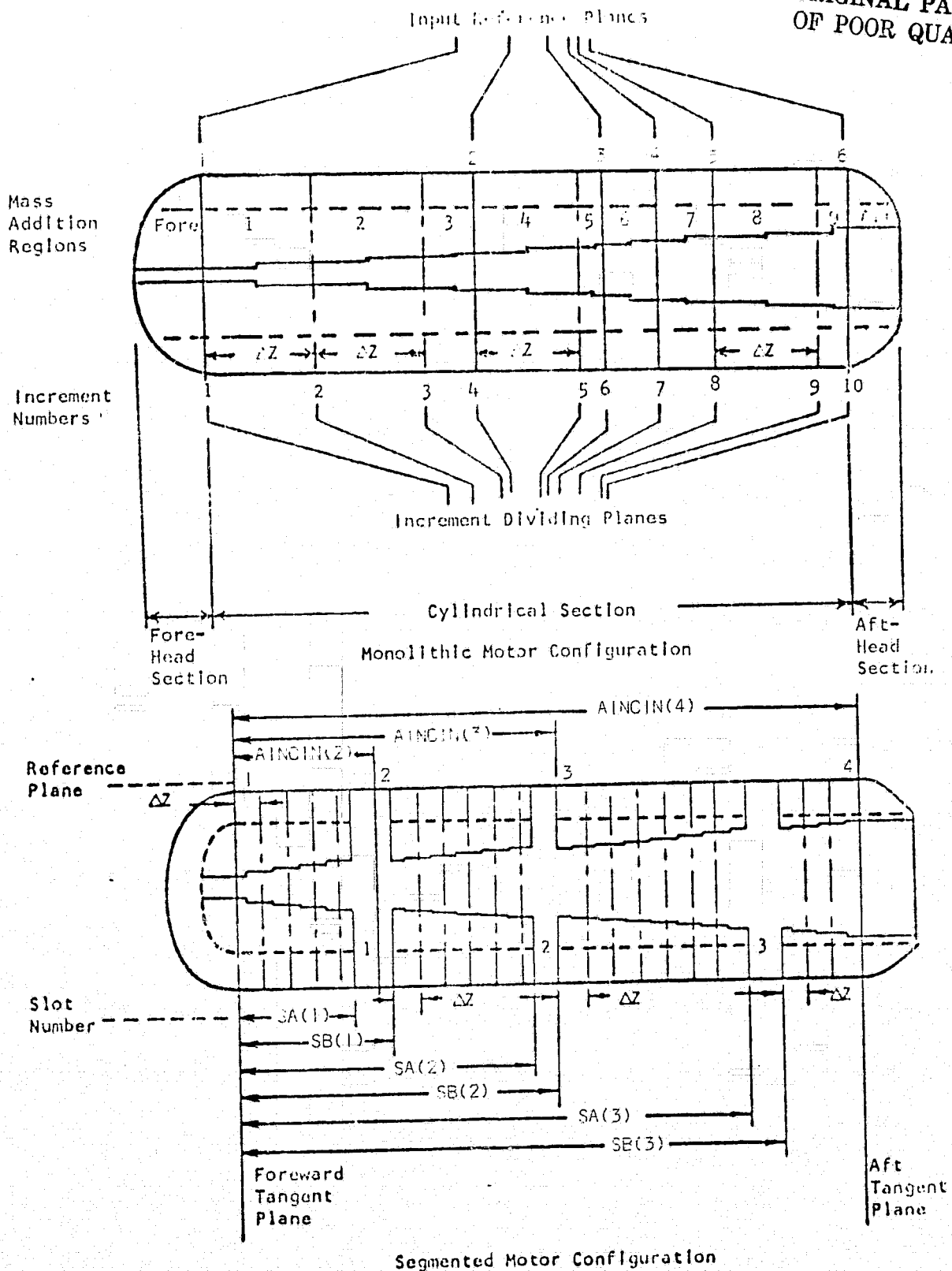


Figure 3.1 Reference Plane and Increment Dividing Plane Identification

USE FOR TYPEWRITTEN MATERIAL ONLY

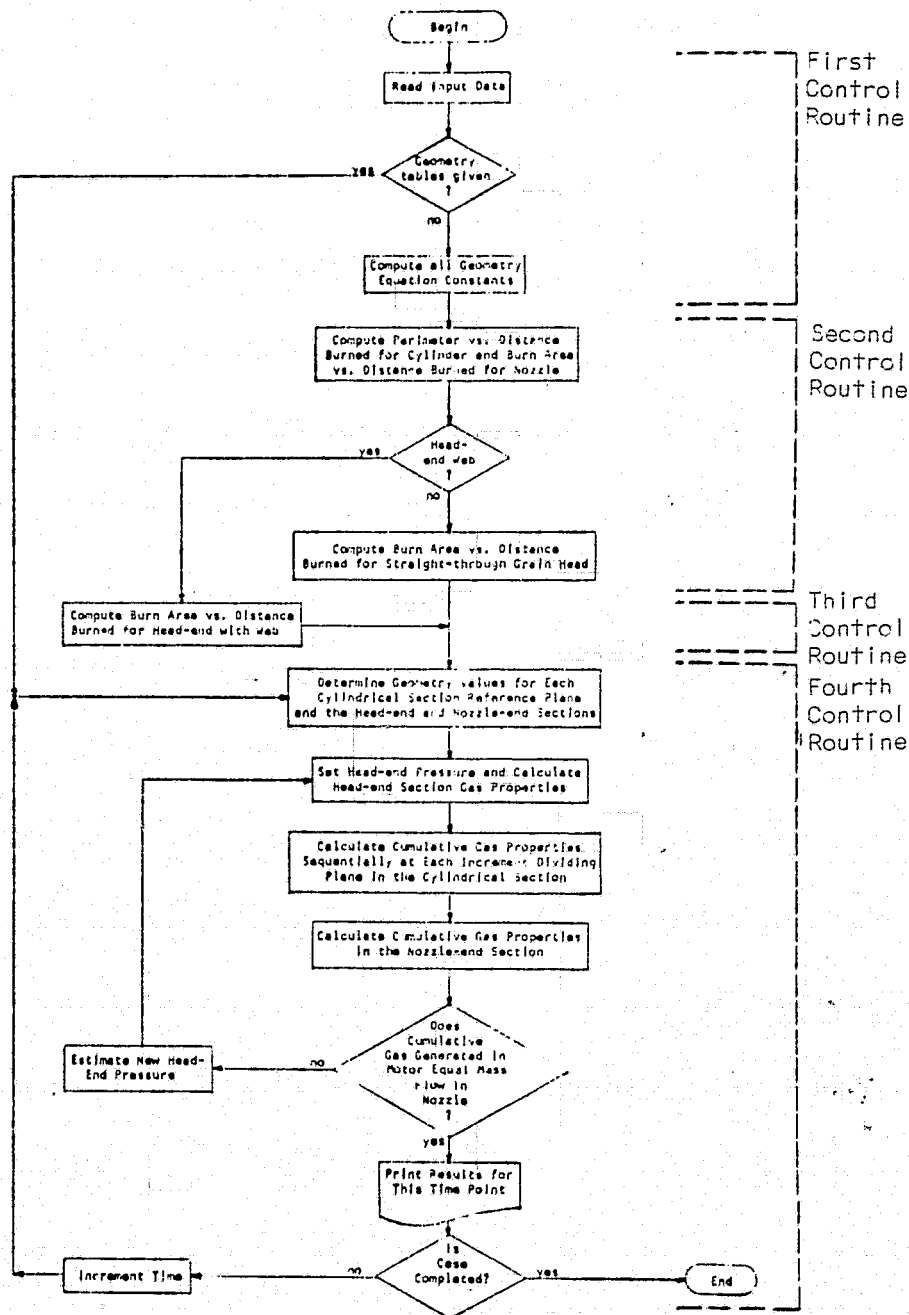


Figure 3.2 Macroscopic Flow Chart

4.0

GAS DYNAMIC EQUATION DEVELOPMENT

The static and total pressure, temperature, velocity, and flow rate of the gas along the length of the propellant grain and through the nozzle are required to determine the progression of the propellant burning surface and motor performance parameters such as chamber pressure, thrust, and total impulse. These parameters are obtained from an iterative solution of the perfect gas law and the equations of continuity, momentum, and energy for one-dimensional gas flow. The gas flow along the propellant grain is determined by dividing the grain into a number of increments which are termed mass addition regions. Two control volumes are shown in Figures 4.1 and 4.2 which define the mass addition regions for the monolithic and segmented motor configurations shown in Figure 4.3. The gas dynamic equations, which are solved for each mass addition region along the propellant grain, are described in the following sections. The isotropic and erosive burning rate equation, anisotropic mass generation model, and propellant gas properties required for the calculations are also described.

4.1

Gas Dynamics for Incremental Control Volumes

The solution of the continuity, momentum, and energy equations for the mass addition regions along the propellant grain and through the nozzle are described in this section. Steady flow, non-steady flow, or non-steady flow with acceleration may be selected.

The following assumptions are basic to the development of the gas dynamic equations:

1. Mass addition occurs as an instantaneous process with no velocity component parallel to the motor axis ($dZ/dt = 0$).
2. The products of combustion obey the perfect gas law.
3. The gas flow is one-dimensional and adiabatic.
4. The friction forces of the combustion gases in the port cavity are negligible.
5. The static pressure is constant across the fore-head section, i.e., no static pressure loss resulting from mass addition or area change.
6. The port area and perimeter are constant across the aft-head section from the aft tangent plane to the grain exit, i.e., constant area duct.

USE FOR TYPEWRITTEN MATERIAL ONLY

4.1

Gas Dynamics for Incremental Control Volumes (Continued)

The control volume or mass addition region Inlet is defined as station 1 and outlet is defined as station 2 as shown in Figures 4.1 and 4.2. The analysis of the mass addition regions along the grain segments is accomplished in subroutines AIBSUB and AIBST to obtain the solution for the discharge pressure, temperature, and flow rate from known Inlet conditions at station 1 and a known value of the instantaneous mass generation rate. The instantaneous mass generation rate is determined from the port perimeters, upstream burning rates, and mass addition region incremental length as follows:

$$\dot{dW} = \frac{(L_{p1} R_{b1} + L_{p2} R_{b2})}{2} \Delta Z \rho_f$$

where

- \dot{dW} = mass generation rate, lb/sec
- L_{p1} = port perimeter at station 1, in
- L_{p2} = port perimeter at station 2, in
- ΔZ = mass addition region length, in
- ρ_f = solid propellant density, lb/in³
- R_{b1} = burning rate at station 1, in/sec
(determined from conditions at adjacent upstream increment dividing plane)
- R_{b2} = burning rate at station 2, in/sec
(determined from conditions at adjacent upstream increment dividing plane)

A mass balance for the slots between the grain segments is obtained in subroutine SLOT. The instantaneous mass generation rate at each slot interface is determined as follows:

$$\dot{dW} = A_f \rho_f a p^n$$

where

- A_f = burning area at slot interface, in²
- p = slot interface static pressure, lb/in²
- a = slot burn rate coefficient
- n = slot burn rate coefficient pressure exponent

The general energy equation and the perfect gas law are applied at station 2 to obtain the state properties:

4.1

Gas Dynamics for Incremental Control Volumes (Continued)

$$U = \left[\frac{2 g_o R \gamma (T_o - T)}{\gamma - 1} \right]^{1/2} \quad (\text{general energy})$$

$$P = 12 \rho R T \quad (\text{perfect gas})$$

The continuity and momentum equations are iterated to obtain the discharge pressure, P_2 , and flow rate, \dot{W}_2 , for each mass addition region. For steady flow conditions, the basic equations are as follows:

$$12 \rho_1 A_1 U_1 + d\dot{W} = 12 \rho_2 A_2 U_2 \quad (\text{continuity})$$

$$\begin{aligned} P_1 A_1 + \frac{P_1 + P_2}{2} (A_2 - A_1) - P_2 A_2 \\ = \frac{12 \rho_1 A_1 U_1}{g_o} (U_2 - U_1) + \frac{U_2 d\dot{W}}{g_o} \quad (\text{momentum}) \end{aligned}$$

For non-steady flow conditions, the basic equations are as follows:

$$\dot{W}_2 = \dot{W}_1 + d\dot{W} - \frac{dW}{dt} \quad (\text{continuity})$$

$$\frac{dW}{dt} = \frac{\partial}{\partial t} \left[\left(\frac{P_1 + P_2}{T_1 + T_2} \right) \frac{V}{12R} \right] \quad (\text{perfect gas})$$

$$\begin{aligned} \frac{\partial}{\partial t} \int_1^2 \frac{(\rho UA)}{g_o} dX + \frac{(\rho_2 U_2^2 A_2 - \rho_1 U_1^2 A_1)}{g_o} \\ = P_1 A_1 - P_2 A_2 + \frac{P_1 + P_2}{2} (A_2 - A_1) \quad (\text{momentum}) \end{aligned}$$

where

P = static pressure, lb/in²

U = velocity, ft/sec

\dot{W} = flow rate, lb/sec

USE FOR TYPEWRITTEN MATERIAL ONLY

4.1

Gas Dynamics for Incremental Control Volumes (Continued)

- T = static temperature, °R
A = port cross sectional area, in²
V = port volume of mass addition region, in³
ρ = density, lb/in³
R = gas constant, °R/ft

4.1.1

Steady Flow Gas Dynamics

The original program developed by the Thiokol Chemical Corporation (references 1 and 2) based the internal gas dynamics on steady flow. Continuity and momentum equations were iterated to obtain mass addition region discharge pressure, temperature, and velocity. This capability is retained in subroutine AIBSUB. The steady flow solution for a segment slot is obtained in subroutine SLOT by neglecting the time dependent terms of the non-steady flow equations. Therefore the steady flow option can be exercised for both monolithic and segmented grain motor designs.

The solution of the steady flow gas dynamics is obtained from an iterative solution of the continuity and momentum equations with the general energy equation and perfect gas law applied at the discharge section to obtain the density and temperature of the combustion gases. The discharge flow rate is obtained from the sum of the mass generation rate and inlet mass flow. An initial value of the discharge velocity is then obtained from the momentum equation, assuming $P_2 = P_1$, and a starting value of the discharge density is obtained from the continuity equation. With the initial value of the discharge density, the following iterative procedure is employed to converge the steady flow equations.

Initially, a guess of the discharge velocity U_2 is obtained from the momentum equation by assuming no pressure loss, $P_2 = P_1$, as follows:

$$\frac{12 \rho_1 A_1 U_1}{g_o} (U_2 - U_1) + \frac{U_2 d\dot{W}}{g_o} = 0$$

or

$$\frac{\dot{W}_1 U_2}{g_o} - \frac{\dot{W}_1 U_1}{g_o} + \frac{U_2 d\dot{W}}{g_o} = 0$$

4.1.1

Steady Flow Gas Dynamics (Continued)

and

$$\dot{W}_1 U_1 = \dot{W}_1 U_2 + U_2 d\dot{W} = \dot{W}_2 U_2$$

From the perfect gas law and continuity equation:

$$U_2 = \frac{\dot{W}_1 U_1}{\dot{W}_2} = \frac{\dot{W}_1 U_1}{\rho_2 A_2 U_2} = \frac{\dot{W}_1 U_1}{\frac{P_2 A_2 U_2}{12 RT_2}}$$

or

$$U_2^2 = \dot{W}_1 U_1 \frac{12 RT_2}{P_2 A_2}$$

Using the general energy equation:

$$\frac{2g_o R \gamma (T_o - T_2)}{\gamma - 1} = \dot{W}_1 U_1 \frac{12 RT_2}{P_2 A_2}$$

Multiplying both sides of the above equation by $-2A_2$ and adding like terms to eliminate T_2 yields:

$$\begin{aligned} \frac{-2g_o R \gamma (T_o - T_2)}{\gamma - 1} 2A_2 + (A_1 + A_2)g_o R T_o - (A_1 + A_2)g_o R T_o \\ + (A_1 + A_2)g_o R T_2 - (A_1 + A_2)g_o R T_2 = \frac{-2A_2 \dot{W}_1 U_1 R T_2}{P_2 A_2} \end{aligned}$$

Rearranging and combining terms yields:

$$\left[\frac{-2g_o \gamma (T_o - T_2)}{\gamma - 1} 2A_2 + (A_1 + A_2)g_o R (T_o - T_2) \right] + \left[(A_1 + A_2)g_o R T_2 + \frac{2A_2 \dot{W}_1 U_1 R T_2}{P_2 A_2} \right] - (A_1 + A_2)g_o R T_o = 0$$

Multiplying through by \dot{W}_2 and again rearranging terms yields:

4.1.1 Steady Flow Gas Dynamics (Continued)

$$\dot{W}_2 \frac{2g_o \gamma (T_o - T_2)}{\gamma - 1} \left[-2A_2 + \frac{(\gamma - 1)(A_1 + A_2)}{2\gamma} \right] + \left[(A_1 + A_2) g_o R T_2 + \frac{2A_2 \dot{W}_1 U_1 R T_2}{P_2 A_2} \right] \dot{W}_2 - (A_1 + A_2) g_o R T_o \dot{W}_2 = 0$$

The terms in the second set of brackets may be rearranged with the perfect gas law, remembering that $P_2 = P_1$:

$$\dot{W}_2 \frac{2g_o \gamma (T_o - T_2)}{\gamma - 1} \left[-2A_2 + \frac{(\gamma - 1)(A_1 + A_2)}{2\gamma} \right] + \left[(A_1 + A_2) g_o P_1 A_2 + 2A_2 \dot{W}_1 U_1 \right] U_2 - (A_1 + A_2) g_o R T_o \dot{W}_2 = 0$$

Thus, from the general energy equation:

$$-\dot{W}_2 U_2^2 \left[2A_2 - \left\{ \frac{\gamma(A_1 + A_2)}{2\gamma} - \frac{(A_1 + A_2)}{2\gamma} \right\} \right] + \left[(A_1 + A_2) g_o A_2 P_1 + 2A_2 \dot{W}_1 U_1 \right] U_2 - (A_1 + A_2) T_o R g_o \dot{W}_2 = 0$$

The above equation is solved for U_2 using the quadratic formula as follows:

$$CAA = -\dot{W}_2 \left[2A_2 - \left\{ \frac{\gamma(A_1 + A_2)}{2\gamma} - \frac{(A_1 + A_2)}{2\gamma} \right\} \right]$$

$$CBB = (A_1 + A_2) g_o A_2 P_1 + 2A_2 \dot{W}_1 U_1$$

$$CC = -(A_1 + A_2) T_o R g_o \dot{W}_2$$

$$RAD = CBB^2 - (4CC) CAA$$

$$Temp_6 = -1 + \frac{RAD^{1/2}}{CBB}$$

$$U_{Tmp} = \frac{CBB Temp_6}{2 CAA}$$

USE FOR TYPEWRITTEN MATERIAL ONLY

4.1.1 Steady Flow Gas Dynamics (Continued)

where U_{Tmp} is based on the assumption of $P_2 = P_1$ and is used only once in calculating the initial value of ρ_{tmp} .

The continuity and momentum equations are iterated by converging on the discharge density as follows:

$$1. \rho_{tmp} = \frac{\dot{W}_2}{12 A_2 U_{Tmp}} \quad (\text{continuity})$$

$$\text{where } \dot{W}_2 = \dot{W}_1 + d\dot{W}$$

$$2. U_{Tmp} = \dot{W}_2 / 12 A_2 \rho_{tmp}$$

$$3. T_{Tmp} = T_o - \frac{(\gamma-1) U_{Tmp}^2}{2 g_o R \gamma} \quad (\text{general energy})$$

$$4. P_{Tmp} = P_1 - \frac{2(\dot{W}_2 U_{Tmp} - \dot{W}_1 U_1)}{g_o (A_1 + A_2)} \quad (\text{momentum})$$

$$5. \rho_2 = \frac{P_{Tmp}}{12 R T_{Tmp}} \quad (\text{perfect gas})$$

$$6. Temp = \rho_{tmp} - \rho_2$$

$$7. \text{ If } \frac{|Temp|}{\rho_{Tmp}} \leq .0001 \text{ go to 8, otherwise set } \rho_{tmp} = \rho_2 \text{ and return to 2.}$$

$$8. \begin{aligned} P_2 &= P_{Tmp} \\ T_2 &= T_{Tmp} \\ U_2 &= U_{Tmp} \end{aligned}$$

9. Determine discharge mach number,

USE FOR TYPEWRITTEN MATERIAL ONLY

4.1.1 Steady Flow Gas Dynamics (Continued)

$$M_2 = \left[\frac{2(T_o - T_{Tnp})}{(\gamma-1)T_{Tnp}} \right]^{1/2}$$

4.1.2 Non-Steady Flow Gas Dynamics

The non-steady flow gas dynamic equations were developed to predict ignition and tail-off transients. The non-steady gas flow equations account for mass, momentum, volume, and pressure within the control volume varying with time. The fundamental equations of momentum and continuity along with the perfect gas law are expressed in partial differential form and then integrated across the control volume describing an incremental mass addition region using the technique of finite differences.

The gas dynamic solution for the mass addition regions is obtained by iterating the continuity and momentum equations for the discharge pressure, temperature, and flow rate. When the solution of the above discharge parameters have converged, the discharge values of mach number, density, and total pressure are determined. The derivation of the time dependent momentum equation and discharge flow rate and pressure equations are presented in the following sections for a mass addition region, Figure 4.1, and a segment slot, Figure 4.2.

4.1.2.1 Mass Addition Region

This section develops gas dynamic equations and solutions for a mass addition region as shown in Figure 4.1. The equation development from fundamental engineering principles to obtain the discharge conditions is taken from reference 6 and is presented below. The discharge pressure,

$$P_2 = \left[\frac{\dot{W}_1 U_1}{g_o} - \frac{\dot{W}_2 U_2}{g_o} + P_1 A_1 + \frac{P_1 + P_2}{2} (A_2 - A_1) \right. \\ \left. - \frac{(P_1 + P_2)(U_1 + U_2)(V - V')}{24 g_o R (T_1 + T_2) \Delta t} - \frac{V(U_1 + U_2)(P_1 + P_2 - P_1' - P_2')}{24 g_o R (T_1 + T_2) \Delta t} \right] / A_2 \\ + \left[\frac{(P_1 + P_2)(A_1 + A_2)a \Delta z}{24 R (T_o + T_2) g_o} \right] / A_2$$

4.1.2.1 Mass Addition Region (Continued)

and discharge flow rate,

$$\dot{W}_2 = \dot{W}_1 + d\dot{W} - \frac{(P_1 + P_2)}{12R(T_1 + T_2)} \frac{V - V'}{\Delta t} - \frac{V(P_1 + P_2 - P_1' - P_2')}{12R(T_1 + T_2)\Delta t}$$

where

V = current port volume, in^3

V' = one past time increment volume, in^3

P_1' = inlet past time increment pressure, lb/in^2

P_2' = discharge past time increment pressure, lb/in^2

Δt = time increment, sec

a = vehicle longitudinal acceleration, ft/sec^2

are iterated in subroutine AIBST to obtain the solution of the gas dynamic equations for a mass addition region. Total pressure, static temperature, mach number, and density are then calculated at the discharge station 2.

The derivation of the above equations and method of convergence follows:

Starting with the continuity equation:

$$1. \quad \frac{\partial}{\partial x} (\rho U A) + \frac{\partial}{\partial t} (\rho A) = 0 \quad (\text{continuity equation})$$

Integrating with respect to x between station 1 and 2 gives:

$$2. \quad \rho_2 U_2 A_2 - \rho_1 U_1 A_1 + \frac{\partial}{\partial t} \int_1^2 \rho A \, dx = 0$$

The first two terms are the discharge and inlet mass flow rates, \dot{W}_1 and \dot{W}_2 , respectively. The integral is the rate of change of mass between the stations 1 and 2 and may be evaluated as follows:

$$3. \quad \frac{\partial}{\partial t} \int_1^2 \rho A \, dx = \frac{\partial}{\partial t} \rho_m V - d\dot{W}$$

where

4.1.2.1 Mass Addition Region (Continued)

$$V = \int_1^2 A \, dx, \text{ in}^3$$

$$\rho_m = \text{average density, lbm/ft}^3$$

$$\dot{dW} = \text{mass flow generated in the section, lb/sec}$$

and upon differentiation:

$$4. \quad \frac{\partial}{\partial t} (\rho_m V) = \rho_m \frac{dV}{dt} + V \frac{d\rho_m}{dt}$$

Combining terms and using finite differences with the perfect gas law yields from the continuity equation the solution of the discharge mass flow in terms of the pressures and control volume:

$$5. \quad \dot{W}_2 = \dot{W}_1 + \dot{dW} - \frac{P_1 + P_2}{12R(T_1 + T_2)} \frac{V - V'}{\Delta t} - \frac{V}{12R(T_1 + T_2)} \frac{P_1 + P_2 - P'_1 - P'_2}{\Delta t}$$

Euler's fluid acceleration equation for unit mass is:

$$6. \quad \frac{\partial U}{\partial t} + U \frac{\partial U}{\partial x} = - \frac{g_o}{\rho} \frac{\partial P}{\partial x}$$

Multiplying Euler's equation (6) by ρA and the continuity equation (1) by U gives:

$$7. \quad \rho A \frac{\partial U}{\partial t} + \rho U A \frac{\partial U}{\partial x} = -g_o A \frac{\partial P}{\partial x}$$

and

$$8. \quad U \frac{\partial}{\partial t} (\rho A) + U \frac{\partial}{\partial x} (\rho U A) = 0$$

Adding equations (7) and (8), and combining appropriate terms, yields the one-dimensional momentum equation:

$$9. \quad \frac{\partial}{\partial t} \frac{(\rho U A)}{g_o} + \frac{\partial}{\partial x} \frac{(\rho U^2 A)}{g_o} = -A \frac{\partial P}{\partial x} = - \frac{\partial}{\partial x} (PA) + P \frac{\partial A}{\partial x}$$

USE FOR TYPEWRITTEN MATERIAL ONLY

4.1.2.1

Mass Addition Region (Continued)

Then, integrating equation (9) with respect to x between station 1 and 2 gives:

$$10. \quad \frac{\partial}{\partial t} \int_1^2 \frac{(\rho UA)}{g_o} dx + \frac{(\rho_2 U_2^2 A_2 - \rho_1 U_1^2 A_1)}{g_o} = (P_1 A_1 - P_2 A_2) + \int_1^2 P dA$$

The last integral may be evaluated by defining a mean pressure, P_m , thus:

$$11. \quad \int_1^2 P dA = P_m (A_2 - A_1)$$

The first integral of equation (10) is the rate of change of momentum from nonstationary changes between stations 1 and 2 and may be evaluated by defining a mean density, ρ_m , and a mean velocity, U_m , and integrating with respect to x , thus:

$$12. \quad \frac{\partial}{\partial t} \int_1^2 \frac{(\rho UA)}{g_o} dx = \frac{\partial}{\partial t} \frac{(\rho_m U_m V)}{g_o}$$

and upon differentiation:

$$13. \quad \frac{\partial}{\partial t} \frac{(\rho_m U_m V)}{g_o} = \frac{\rho_m U_m}{g_o} \frac{dV}{dt} + \frac{\rho_m V}{g_o} \frac{dU_m}{dt} + \frac{U_m V}{g_o} \frac{d\rho_m}{dt}$$

Letting $\frac{dU_m}{dt} = 0$ and combining terms using finite differences with the perfect gas law yields from the momentum equation the solution of the discharge pressure in terms of the mass flows and pressures:

$$14. \quad P_2 = \left[\frac{\dot{W}_1 U_1}{g_o} - \frac{\dot{W}_2 U_2}{g_o} + P_1 A_1 + \frac{P_1 + P_2}{2} (A_2 - A_1) - \frac{(P_1 + P_2)(U_1 + U_2)(V - V_1)}{24 g_o R (T_1 + T_2) \Delta t} - \frac{V(U_1 + U_2)(P_1 + P_2 - P_1^1 - P_2^1)}{24 g_o R (T_1 + T_2) \Delta t} \right] / A_2$$

The acceleration term to be added to equation 14 is derived in Section 4.1.3.

4.1.2.1

Mass Addition Region (Continued)

Equations (5) and (14) are iterated in subroutine AIBST to solve for discharge temperature, pressure, and flow rate at each mass addition region as follows:

1. Estimate a starting value of the discharge pressure and temperature using the influence coefficient equations for constant specific heat and molecular weight, reference 7.

$$DT = T_1(\gamma-1) \left[\frac{(1+\gamma M_1^2) M_1^2 d\dot{W}}{(1-M_1^2) \dot{W}_1} \right]$$

$$a) T_{2\text{guess}} = T_1 - DT$$

$$DP = P_1 d\dot{W} \frac{2\gamma M_1^2}{\dot{W}_1 (1-M_1^2)} \left[1 + \frac{(\gamma-1) M_1^2}{2} \right]$$

$$b) P_{2\text{guess}} = P_1 - DP$$

2. Determine the gas storage:

$$\frac{dW}{dt} = \left[\frac{(P_1 + P_{2\text{guess}})(2V - V') - V(P_1' + P_2')}{12 R (T_1 + T_{2\text{guess}}) \Delta t} \right]$$

3. Determine the discharge flow rate:

$$\dot{W}_2 = \dot{W}_1 + d\dot{W} - \frac{dW}{dt}$$

4. Determine the discharge velocity:

$$U_2 = \frac{\dot{W}_2 R T_{2\text{guess}}}{P_2 A_2}$$

USE FOR TYPEWRITTEN MATERIAL ONLY

4.1.2.1 Mass Addition Region (Continued)

5. Determine the discharge pressure (equation 14 above):

$$P_2 = \left\{ \frac{\dot{W}_1 U_1}{g_o} - \frac{\dot{W}_2 U_2}{g_o} + P_1 A_1 + \frac{(P_1 + P_{2\text{guess}})}{2} (A_2 - A_1) \right. \\ \left. - \left[V(U_1 + U_2) (P_1 + P_{2\text{guess}} - P_1^i - P_2^i) \right. \right. \\ \left. \left. + (P_1 + P_{2\text{guess}}) (U_1 + U_2) (V - V^i) \right] \right\} / \left[24 g_o R \right. \\ \left. (T_1 + T_{2\text{guess}}) \Delta t \right] / A_2$$

6. If an accelerating reference system is considered (see Section 4.1.3 for derivation):

If $\frac{a}{g_o} > 0$, determine acceleration term:

$$\text{Temp} = \left[(P_1 + P_{2\text{guess}}) (A_1 + A_2) \frac{a}{g_o} \Delta z \right] / \left[24 R (T_1 + T_{2\text{guess}}) A_2 \right]$$

$$P_2 = P_2 + \text{Temp}$$

7. If
- $\left| \frac{P_2 - P_{2\text{guess}}}{P_{2\text{guess}}} \right| \leq .001$
- , go to step 8, otherwise obtain new value of
- $P_{2\text{guess}}$
- using method of false position and return to step 1(b).

8. Determine
- U_2
- based on converged
- P_2
- :

$$U_2 = \frac{\dot{W}_2 R T_{2\text{guess}}}{P_2 A_2}$$

9. Determine the discharge temperature using the general energy equation:

$$T_2 = T_o - \frac{(\gamma - 1) U_2^2}{2 g_o \gamma R}$$

4.1.2.1

Mass Addition Region (Continued)

10. If $\left| \frac{T_{2\text{guess}} - T_2}{T_{2\text{guess}}} \right| \leq .001$, go to step 11, otherwise obtain new value of $T_{2\text{guess}}$ using method of false position and return to step 1(a).

11. Solution is converged, determine discharge mach number, gas density, and total pressure:

$$M_2 = U_2 / \left[g_o \gamma R T_2 \right]^{1/2}$$

$$\rho_2 = \frac{\dot{W}_2}{12 A_2 U_2}$$

$$P_{02} = P_2 \left(\frac{T_0}{T_2} \right)^{\frac{\gamma}{\gamma-1}}$$

4.1.2.2

Segment Slot Mass Addition

The development of the gas dynamic equations for the region between grain segments of segmented motors (referred to as a slot) is similar to the non-steady gas flow equation development for a mass addition region, Section 4.1. The control volume for a slot is defined from the forward slot interface to the aft slot interface with mass addition occurring at each interface and not within the control volume. The control volume for a slot is shown in Figure 4.2.

The following assumptions are made in subroutine SLOT (including the assumptions of Section 4.1):

- a. The static pressure and temperature at the slot interface is the same as the port static pressure, $P_1 = P_2$, $P_3 = P_4$, $T_1 = T_2$, and $T_3 = T_4$.
- b. The mass flow generated at the slot interface is a function of the port static pressure only and is determined from the following burn rate equation:

$$\dot{dW} = A_f \rho_f a P^n$$

4.1.2.2 Segment Slot Mass Addition (Continued)

where:

- \dot{dW} = generated mass flow, lb/sec
 A_f = burn area at slot interface, in²
 ρ_f = solid propellant density, lb/in³
 P = slot interface static pressure, lb/in²
 a = burn rate coefficient
 n = burn rate equation pressure coefficient.

- c. Static pressure at station 3 is a function of the area change (dA/dx), and the capacitance effects (dP/dt and dV/dt) between stations 2 and 3, and acceleration of the vehicle.

The solution of the gas dynamics within a slot is obtained from the above assumptions and the equations developed in Section 4.0 as follows:

1. Determine the mass generation rate at the forward slot interface (station 2) from the static pressure at station 1 (port cavity discharge)

$$\dot{dW}_f = A_f \rho_f a P^n$$

2. Determine the inlet flow rate, velocity, and mach number:

$$\dot{W}_2 = \dot{W}_1 + \dot{dW}_f$$

$$U_2 = \frac{\dot{W}_2 R T_1}{P_1 A_2}$$

$$M_2 = \frac{U_2}{[g_o k R T_1]^{1/2}}$$

3. Determine the current slot volume and rate of change of volume:

$$V = \frac{A_2 + A_3}{2} (z_3 - z_2)$$

where

ORIGINAL PAGE IS
OF POOR QUALITY

NUMBER D2-125286-1
REV LTR

THE EDI

F COMPANY

4.1.2.2 Segment Slot Mass Addition (Continued)

z_3 = aft interface station location, in.

z_2 = forward interface station location, in.

$$\frac{dV}{dt} = \frac{\pi}{2\Delta t} \left[\tau_f (R_{f2}^2 + R_{SLOTf}^2) + \tau_a (R_{f3}^2 + R_{SLOTa}^2) \right]$$

where

τ_f = last value of forward slot interface distance burned in.

τ_a = last value of aft slot interface distance burned in.

4. Guess the value of the slot aft interface static pressure and temperature using the influence coefficient equations for constant specific heat and molecular weight, reference 7.

$$dT = \frac{T_1 (\gamma - 1) (1 + \gamma M_2^2) M_2^2 d\dot{W}_f}{(1 - M_2^2) \dot{W}_2}$$

$$dP = \frac{P_1 d\dot{W}_f \frac{4}{\gamma} g_o M_2^2 (1 + \frac{\gamma - 1}{2} M_2^2)}{\dot{W}_2 (1 - M_2^2)}$$

$$P_{3\text{guess}} = P_2 - dP$$

$$T_{3\text{guess}} = T_1 - dT$$

5. Determine the mass generation rate at the aft slot interface (station 3) from the guessed static pressure at station 3:

$$\dot{W}_a = \rho_f A_a a P_{3\text{guess}}^n$$

6. Determine the stored gas in the slot

$$\frac{dW}{dt} = \frac{(P_2 + P_{3\text{guess}})}{12 R (T_1 + T_{3\text{guess}})} \frac{dV}{dt} + \frac{V}{12 R (T_1 + T_{3\text{guess}})} \frac{(P_2 + P_{3\text{guess}} - P_2^i - P_3^i)}{\Delta t}$$

USE FOR TYPEWRITTEN MATERIAL ONLY

4.1.2.2 Segment Slot Mass Addition (Continued)

where

$$\frac{dW}{dt} = \frac{\partial}{\partial t} \left(\frac{P V}{12 RT} \right)$$

7. Determine the discharge flow rate and velocity:

$$\dot{W}_3 = \dot{W}_2 + d\dot{W}_a - \frac{dW}{dt}$$

$$U_3 = \frac{\dot{W}_3 R T_{3\text{guess}}}{P_{3\text{guess}} A_3}$$

8. Determine the aft slot interface static pressure (see Section 4.1.2.1, equation 14).

$$P_3 = \left[\frac{\dot{W}_2 U_2}{g_o} - \frac{\dot{W}_3 U_3}{g_o} + P_2 A_2 + \frac{P_2 + P_{3\text{guess}}}{2} (A_3 - A_2) \right. \\ \left. - \frac{(P_2 + P_{3\text{guess}})(U_2 + U_3)}{24 g_o R (T_{3\text{guess}} + T_1)} \frac{dV}{dt} \right. \\ \left. - \frac{V(U_2 + U_3)(P_2 + P_{3\text{guess}} - P_2^i - P_3^i)}{24 g_o R (T_{3\text{guess}} + T_1) \Delta t} \right] / A_3 \\ + \left[\frac{(P_2 + P_{3\text{guess}})(A_2 + A_3) a \Delta Z}{24 g_o R (T_2 + T_{3\text{guess}})} \right] / A_3$$

9. If
- $\left| \frac{P_3 - P_{3\text{guess}}}{P_3} \right| \leq \text{CRP}$
- , go to step 10, otherwise obtain new

value of $P_{3\text{guess}}$ using method of false position and return to step 5. If CRP is not input the program will set CRP equal to .001.

10. Determine the slot aft interface static temperature from the general energy equation:

USE FOR TYPEWRITTEN MATERIAL ONLY

4.1.2.2 Segment Slot Mass Addition (Continued)

$$T_3 = T_0 - \frac{(\gamma - 1) U_3^2}{2 g_0 \gamma R}$$

11. If $\left| \frac{T_3 - T_{3\text{guess}}}{T_3} \right| \leq .\text{CRT}$, go to step 12, otherwise obtain new value of $T_{3\text{guess}}$ using method of false position and return to step 7. If CRT is not input the program will set CRT equal to .001.
12. Determine the slot discharge velocity, mach number, density, and total pressure:

$$U_4 = \frac{\dot{W}_3 R T_3}{P_3 A_4}$$

$$M_4 = \frac{U_4}{[g_0 \gamma R T_3]^{1/2}}$$

$$\rho_4 = \frac{\dot{W}_3}{12 A_4 U_4}$$

$$P_{04} = P_3 \left(\frac{T_0}{T_3} \right)^{\frac{\gamma}{\gamma-1}}$$

4.1.3 Non-Steady Flow Gas Dynamics with Acceleration

The effects of longitudinal acceleration of the vehicle on the internal ballistic solution is considered in this section. In conventional gas dynamic studies, the effects of gravitational forces are not considered because, for compressible fluids, gravitational forces are significantly less than surface forces. Recent development of missiles for low level ICBM intercept may require boost accelerations that are of significant magnitude to affect motor internal pressures and temperatures. In an accelerating reference system, the force field which results from the acceleration is equivalent to a gravitational force field in a nonaccelerating reference system.

USE FOR TYPEWRITTEN MATERIAL ONLY

4.1.3

Non-Steady Flow Gas Dynamics with Acceleration

The acceleration effects are considered only on the gas dynamic equations in the port cavity of the motor and not in the nozzle. Effect of acceleration on a nozzle is to move the sonic point upstream of the throat. The effect of acceleration on the motor internal ballistics is to reduce the pressure drop along the propellant grain as well as fore-head pressure.

The resulting acceleration term is added to the momentum equation as follows:

$$\frac{\partial}{\partial t} \int_1^2 \frac{(\rho U A)}{g_o} dx + \frac{(\rho_2 U_2^2 A_2 - \rho_1 U_1^2 A_1)}{g_o}$$

$$= (P_1 A_1 - P_2 A_2) + \int_1^2 P dA + \frac{(\rho_m A_m a \Delta Z)}{g_o}$$

where

- a = vehicle longitudinal acceleration, ft/sec²
- ΔZ = length between increments, ft
- ρ_m = average gas density in increment, lb/ft³
- A_m = average cross sectional area in increment, ft²
- g_o = conversion constant, 32.174 lbm/slug

The acceleration term is developed as follows:

1. From Newton's Second Law of motion:

$$F_{bf} = \frac{W}{g_o} a$$

where

- F_{bf} = body force, lbf
- W = weight, lbm
- a = acceleration, ft/sec²

4.1.3

Non-Steady Flow Gas Dynamics with Acceleration

2. The body force exerted on the gas within a mass addition region is:

$$F_{bf} = \frac{\rho_m A_m \Delta Z}{g_o} a$$

3. Assuming perfect gas relationships, the body force may be written as:

$$F_{bf} = \frac{(P_1 + P_2)(A_1 + A_2) \Delta Z a}{24 g_o R (T_1 + T_2)}$$

4. and from $P = F/A$:

$$\Delta P = \frac{(P_1 + P_2)(A_1 + A_2) \Delta Z}{24 R (T_1 + T_2) A_2} \frac{a}{g_o}$$

where

ΔP = pressure change resulting from acceleration, lb/in²

The pressure change resulting from vehicle acceleration is added to the discharge pressure of a mass addition region as shown in Section 4.1.2.1, step 6, for the iteration of the discharge pressure and flow rate.

USE FOR TYPEWRITTEN MATERIAL ONLY

4.2 Complete Motor Gas Dynamics

Solution of overall motor gas dynamics or internal ballistics from fore-head to nozzle exit is described in this section.

4.2.1 Fore-Head Pressure Convergence

Solution of motor internal ballistics for each time point is obtained by an iteration process which converges on fore-head pressure, PH, (Figure 3.1). An initial estimate is made for fore-head pressure either from the input value PH1 at time = 0, or from the previous time solution of PH at time > 0. The fore-head section mass balance is obtained in subroutine MNCHN4 (flow chart No. 10) after geometry values have been determined. Then parameters necessary to solve cylindrical section mass addition regions in subroutine SEGSUB (flow chart number 11) are determined. When the cylindrical section is complete, the aft-head section mass balance is obtained in subroutine MNCHN4 and the fore-head pressure is checked for convergence in subroutine SETPH (flow chart number 12).

The fore-head pressure convergence check in subroutine SETPH is made as follows:

1. Determine the throat critical pressure ratio:

$$\frac{P_s}{P_0} = \left(\frac{2}{\gamma+1} \right)^{\frac{\gamma}{\gamma-1}}$$

2. Determine the nozzle total pressure:

$$P_{ON} = P \left(\frac{T_0}{T} \right)^{\frac{\gamma}{\gamma-1}}$$

where

P = aft-head section discharge static pressure,
lbs/in²

T = aft-head section discharge static temperature, °R

3. If $P_{ON} \left(\frac{P_s}{P_0} \right) \geq P_a$, the nozzle is choked. Determine the sonic nozzle flow rate:

4.2.1 Fore-Head Pressure Convergence (Continued)

$$DIS = \frac{g_o NN A_t P_{OH}}{C^*}$$

where

NN = number of nozzles

Go to 5.

4. If $P_{ON} \left(\frac{P_s}{P_0} \right) < P_a$, the nozzle is not choked. Determine the subsonic nozzle flow rate:

$$SDIS = NN P_{ON} A_{EE} \left(\frac{P_a}{P_{ON}} \right)^{\frac{1}{\gamma}} \left[\frac{2\gamma g_o}{T_o R (\gamma-1)} \left(1 - \left(\frac{P_a}{P_{ON}} \right)^{\frac{\gamma-1}{\gamma}} \right) \right]^{1/2}$$

where

A_{EE} = nozzle exit plane area.

DIS = SDIS

5. If $\left| \frac{\dot{W} - DIS}{DIS} \right| > CRW$, estimate new fore-head pressure as follows (\dot{W} = grain discharge flow rate). If CRW is not input the program will set CRW equal to .001.

a) $WD = \frac{\dot{W} - WDB}{P_H - P_{Hold}}$

where P_{Hold} = previous iterative value of P_H , psi

WDB = previous iterative value of \dot{W} , lb/sec

\dot{W} = grain discharge flow rate, lb/sec

b) $DEED = \frac{DIS - DISB}{P_H - P_{Hold}}$

where $DISB$ = previous iterative value of DIS

USE FOR TYPEWRITTEN MATERIAL ONLY

4.2.1

Fore-Head Pressure Convergence (Continued)

- c) If
- $DEED = WD$
- or if
- $DEED = 0$
- :

$$\Delta P = P_H \left[\left(\frac{\dot{W}}{DIS} \right)^{1.4} - 1 \right]$$

- d) If
- $DEED \neq WD$
- :

$$\Delta P = \frac{\dot{W} - DIS}{DEED - WD}$$

- e)
- $WDB = \dot{W}$

$$DISB = DIS$$

- f) If
- $\dot{W} \leq DIS$
- ,
- $P_{min} = P_H$

- g) If
- $\dot{W} > DIS$
- ,
- $P_{max} = P_H$

- h)
- $P_{Hguess} = P_H + \Delta P$

- i) If
- $P_{Hguess} \leq P_{min}$
- and
- $P_{min} = 0$

$$P_{Hguess} = 5.0 \text{ lbs/in}^2$$

- j) If
- $P_{Hguess} \leq P_{min}$
- and
- $P_{min} \neq 0$

$$P_{Hguess} = 2.0 P_H$$

- k) If
- $P_{min} < P_{Hguess} \leq P_{max}$
- :

$$P_{Hguess} = .9(P_{max} - P_{min}) + P_{min}$$

Return to fore-head section with $P_H = P_{Hguess}$.

USE FOR TYPEWRITTEN MATERIAL ONLY

4.2.1 Fore-Head Pressure Convergence (Continued)

- 1) If $P_{\min} < P_{\text{Hguess}} < P_{\max}$, return to fore-head section with $P_H = P_{\text{Hguess}}$.
6. If $\left| \frac{\dot{W} - \text{DIS}}{\text{DIS}} \right| \leq \text{CRW}$, convergence has been attained. If CRW is not input the program will set CRW equal to .001.

4.2.2 Nozzle Gas Dynamics

After the fore-head pressure convergence criterion has been satisfied (step 5 and 6 of the previous section), the nozzle gas dynamics in subroutine SETPH are determined as follows:

1. Determine the nozzle exit area:

$$A_{EE} = \frac{\pi D_E^2}{4}$$

2. Determine the nozzle expansion ratio:

$$\epsilon_G = \frac{A_{EE}}{A_t}$$

3. Iterate the following equation for $\frac{P_E}{P_{ON}}$ using the method of false position:

$$\frac{P_E}{P_{ON}} = \left[\left(\frac{\gamma-1}{\gamma+1} \right)^{1/2} \left(\frac{2}{\gamma+1} \right)^{\frac{1}{\gamma-1}} / \epsilon_G \right] / \left[1 - \left(\frac{P_E}{P_{ON}} \right)^{\frac{\gamma-1}{\gamma}} \right]^{\frac{\gamma}{2}}$$

4. Determine the momentum portion of the thrust coefficient:

$$C_{fo} = \left[\frac{2\gamma^2}{\gamma-1} \left(\frac{2}{\gamma+1} \right)^{\frac{\gamma+1}{\gamma-1}} \left\{ 1 - \left(\frac{P_E}{P_{ON}} \right)^{\frac{\gamma-1}{\gamma}} \right\} \right]^{1/2}$$

4.2.2

Nozzle Gas Dynamics (Continued)

5. Determine the delivered vacuum thrust coefficient:

$$C_{foL} = (C_{fo} \lambda_n + \frac{P_E}{P_{ON}} \epsilon_G) C_m$$

6. Determine the delivered thrust at
- P_a
- for sonic flow:

$$F = (C_{foL} P_{ON} A_t - P_a \epsilon_G A_t) NN$$

7. Determine the delivered thrust for subsonic flow:

$$V_E = \left[\frac{2\gamma T_o R g_o}{\gamma-1} \left\{ 1 - \left(\frac{P_a}{P_{ON}} \right)^{\frac{\gamma-1}{\gamma}} \right\} \right]^{1/2}$$

$$F = \frac{V_E \lambda_N \dot{W} C_m NN}{g_o}$$

8. Determine the fore-head pressure-time integral:

$$\int P_H dt = \int P_H dt + (P_{Hold} + P_H) \frac{\Delta t}{2}$$

9. Determine the nozzle total pressure-time integral:

$$\int P_{ON} dt = \int P_{ON} dt + (P_{ONold} + P_{ON}) \frac{\Delta t}{2}$$

10. Determine the nozzle discharge flow-time integral:

$$\int \dot{W}_N dt = \int \dot{W}_N dt + (\dot{W}_{Nold} + \dot{W}_N) \frac{\Delta t}{2}$$

11. Determine the total impulse:

$$IT = IT + (F + F_{old}) \frac{\Delta t}{2}$$

USE FOR TYPEWRITTEN MATERIAL ONLY

ORIGINAL PAGE IS
OF POOR QUALITY

THE **DRB** COMPANY

NUMBER D2-125206-1
REV LIR

4.2.2 Nozzle Gas Dynamics (Continued)

The above iterative procedure is allowed to continue for no more than 21 iterations. If the fore-head pressure has not converged within 21 iterations, a summary of the last iteration is printed followed by a complete program data dump and the next case is set up. When convergence has been attained, the program output is printed and the next time increment solution is initiated by setting P_{Hguess} equal to the previous time increment solution of P_H . Program execution for each case continues until termination options are satisfied.

4.3 Propellant Characteristics and Burning Rate Model

The characteristics of the propellant are represented by a mathematical model of the burning rate and properties of the products of combustion. Basic assumptions of a perfect gas, constant combustion temperature and constant specific heat allow the propellant gas properties to be described by the characteristic velocity, C^* , the combustion temperature, T_o , the specific heat ratio, γ , and the gas constant, R .

The propellant burning rate model allows either isotropic or anisotropic burning of the propellant surface. Isotropic burning is defined as uniform combustion occurring normal to the propellant surface. In anisotropic burning, the burn rate varies with distance burned as well as with conditions at the gas-propellant interface. It results from non-homogeneous dispersion of propellant additives near the case wall and core during propellant casting.

4.3.1 Propellant Gas Properties

The propellant gas properties may be held constant or may be varied as a function of the static pressure in the port cavity. If the gas properties are to be held constant, the parameters T_o , C^* , γ , and R are input. If the gas properties are to be varied, tables of the combustion temperature (TCOMB), the molecular weight (AMWG), the specific heat ratio (GAMAG), and the characteristic velocity (CSTR), are input as a function of static pressure (PRESS).

When the gas tables are input, a linear interpolation procedure is used to obtain the gas properties for the pressure at the increment dividing plane.

USE FOR TYPEWRITTEN MATERIAL ONLY

4.3.2 Propellant Burning Rate Model

The propellant burning rate model may include the effects of erosive burning. Erosive burning is defined as the change in the local burning rate resulting from gas velocity parallel to the burning surface.

The propellant burning rate, R_B , at any increment dividing plane is determined from the following parameters at the adjacent upstream increment dividing plane:

- | | |
|-----------------------------------|-----------|
| 1. Static pressure | P |
| 2. Gas velocity | U |
| 3. Mass velocity per unit area | G |
| 4. Distance from stagnation point | h_{RB} |
| 5. Burning rate | R_{BHI} |
| 6. Solid propellant density | ρ_f |

With the exception of ρ_f , these values are not input but are calculated within the program.

Fifty-one constants are available to define the burning rate equation. Only the constants that are required for the particular burning rate equation to be used are input. These constants are as follows:

KG1 through KG5
KU1 through KU5
KR1 through KR39
KSL0T1 through KSL0T2

Prior to calculating burning rate, critical values of velocity, UCR, and mass velocity per unit area, GCR, are obtained as follows:

$$UCR = KU1 + (KU2) P^{(KU3)} + (KU4) P^{(KU5)}$$

$$GCR = KG1 + (KG2) P^{(KG3)} + (KG4) P^{(KG5)}$$

Propellant burning rate is then calculated in one of two ways:

If G is greater than or equal to GCR , and U is greater than or equal to UCR , the following relation is used:

$$R_B = KR1 + (KR2) P^{(KR3)} + (KR4) P^{(KR5)} + (KR6) U^{(KR7)} \\ + (KR8) U^{(KR9)} + (KR10) G^{(KR11)} + (KR12) G^{(KR13)}$$

4.3.2

Propellant Burning Rate Model

$$\begin{aligned}
 & + (KR14) P^{(KR15)} U^{(KR16)} + (KR17) P^{(KR18)} U^{(KR19)} \\
 & + (KR20) P^{(KR21)} G^{(KR22)} + (KR23) P^{(KR24)} G^{(KR25)} \\
 & + \frac{KR26}{(KR27) P^{(KR28)} + (KR29) P^{(KR30)}} \\
 & + (KR31) \frac{G^{(KR32)}}{h_{RB}^{(KR33)}} e^{-\left[\frac{(KR34) R_{BHI} P_f}{G} \right]}
 \end{aligned}$$

If the value of G is less than GCR or if the value of U is less than UCR, then the following relation is used:

$$R_B = KR35 + (KR36) P^{(KR37)} + (KR38) P^{(KR39)}$$

To prevent the values of GCR and UCR from being used simultaneously for choosing the burning rate model, one of these values must always be equal to zero. This is accomplished by setting the values of constants KG1, KG2, and KG4 or the values of constants KU1, KU2, and KU4 equal to zero. The program will automatically stop if any of the terms KG1, KG2, or KG4 are not equal to zero when any of the terms KU1, KU2, or KU4 are also not equal to zero. In such cases, the program will print-out a statement that the GCR or UCR coefficients are invalid. If KR26 is not zero, then neither KR27 nor KR29 can be negative or simultaneously equal to zero; if this restriction is exceeded, the program will stop and print-out a statement of invalid KR27 or KR29.

The burning rate at the segment slot face is calculated by

$$RBSLOT = (KSL0T1) P^{(KSL0T2)}$$

where P is the static pressure in the port at the segment interface.

USE FOR TYPEWRITTEN MATERIAL ONLY

4.3.3

Anisotropic Propellant Burning

Anisotropic propellant burning capability, where burn rate depends on distance burned, was added to the program because of Boeing's experience with the HiBEX motor. Anisotropic burning occurred during both ignition and tail-off. It appeared to be the result of two effects: 1) variation in the alignment of the staples between the bulk of the propellant at the case wall and the core interface, and 2) the burning distance required to develop "coning" about the staples. Anisotropic burning is most easily represented by variation of the constant "a" as a function of distance burned in the burn rate equation, $r = aP^n$. During ignition, mass generation is determined by multiplying the port perimeter by the anisotropic burning rate. During tail-off, regions exist where both isotropic and anisotropic burning occur. The port perimeter is subdivided accordingly. The total mass generation is then the sum of the individual mass generation rates.

The following assumptions have been made in developing the mathematical model:

1. The anisotropic region at both the core interface and the case wall is of uniform thickness along the motor length.
2. The thickness of the anisotropic region is the same at both the core interface and the case wall.
3. The burn rate variation through the anisotropic region is a function only of distance burned and local static pressure, $r = a(\tau)P^n$.
4. The anisotropic burn rate increases from the core interface toward the isotropic propellant and decreases from the isotropic propellant toward the case wall.
5. The fore-head and aft-head burning rate during motor tail-off is the same as the adjacent tangent plane isotropic burning rate.

The following limitations apply to the program:

1. Anisotropic burning cannot be considered for propellants with wagon wheel grain configurations during tail-off.
2. Anisotropic burning may be considered for non-steady flow options only during ignition and tail-off.

Program simulation of anisotropic burning is accomplished by altering normal program solution during the ignition transient interval to solve for the burn rate coefficient with a fixed

USE FOR TYPEWRITTEN MATERIAL ONLY

4.3.3 Anisotropic Propellant Burning (Continued)

value of the fore-head pressure at each time increment. The burn rate coefficient is stored in a table as a function of distance burned at a desired location within the cylindrical section (Input NINCPL). A program option is available to input this anisotropic burn rate coefficient table and solve for fore-head pressure as discussed in Section 4.2.1. The burning rate within the fore-head and aft-head sections may be specified by inputs KRH and KRN or by the anisotropic burning rate table inputs. During the tail-off interval, when the burning surface is within the anisotropic propellant region, the burning rate at each increment dividing plane in sectors 6, 7, and 8 becomes a function of the distance from the case wall within the sector as shown in Figures 4.4, 4.5, and 4.6.

The program method of solution for anisotropic burning during the ignition transient interval, when the fore-head pressure trace is input, is altered to converge on the burn rate coefficient, KRST. At each time increment, an estimate of the burn rate coefficient is determined in subroutine RBSTSB from the fore-head pressure rise rate and the motor configuration as follows:

From the perfect gas law using finite differences:

$$1. \quad \frac{dP_H}{dt} = \frac{\partial}{\partial t} \left[\frac{12 W_H R T}{V} \right] = \frac{12 R T}{V} \frac{dW_H}{dt}$$

and,

$$2. \quad \frac{dW_H}{dt} = \dot{W}_{in} - \dot{W}_{out}$$

where

$$\begin{aligned} \dot{W}_{in} &= \text{generated weight flow rate, lb/sec} \\ \dot{W}_{out} &= \text{nozzle discharge flow rate, lb/sec} \\ V &= \text{free volume, in}^3 \end{aligned}$$

The nozzle discharge flow rate is determined from the nozzle geometry:

$$3. \quad \dot{W}_{out} = \frac{g_o A_t P_{ON}}{C^*}$$

where

4.3.3

Anisotropic Propellant Burning (Continued)

$$\begin{aligned} g_o &= \text{gravitational constant, ft/sec}^2 \\ A_t &= \text{nozzle throat area, in}^2 \\ P_{ON} &= \text{nozzle total pressure, lb/in}^2 \\ C^* &= \text{characteristic velocity, ft/sec}^2 \end{aligned}$$

ORIGINAL PAGE IS
OF POOR QUALITY

and the generated weight flow rate is determined from the motor configuration:

$$4. \dot{W}_{in} = R_b A_b \rho_f$$

where

$$\begin{aligned} R_b &= \text{burn rate, in/sec} \\ A_b &= \text{total burn area, in}^2 \\ \rho_f &= \text{propellant density, lb/in}^3 \end{aligned}$$

$$5. \Gamma^2 = \frac{\Gamma^2 C^{*2}}{g_o}$$

where

$$\Gamma^2 = \gamma \left(\frac{2}{\gamma+1} \right)^{\frac{\gamma+1}{\gamma-1}}$$

Therefore:

$$6. \frac{dP_H}{dt} = \dot{P}_H = \frac{12 \Gamma^2 C^{*2}}{V g_o} (R_b A_b \rho_f) - \frac{12 \Gamma^2 C^{*2}}{V} (P_{ON} A_t)$$

Combining and arranging terms with $P_{ON} = (\text{TPR}) P_H$, where TPR is an estimate of the port pressure drop, we have:

$$7. R_b = \frac{\dot{P}_H + \frac{(12 \Gamma^2 C^{*2}) (A_t \text{TPR} P_H)}{V}}{\frac{12 \Gamma^2 C^{*2} A_b \rho_f}{g_o V}}$$

and

4.3.3

Anisotropic Propellant Burning (Continued)

$$8. \quad KRST = \frac{R_b}{P_H^n}$$

After the initial estimate of KRST is made, the ballistic solution is converged for the fixed fore-head pressure obtained from the input pressure trace using the convergence procedure outlined in Section 4.2.1. When the anisotropic burn rate coefficient table is input, the method of solution remains unchanged except that the burn rate coefficient which depends on distance burned at location NINCPL from the forward tangent plane is determined from the input table at each time increment.

The ignition transient interval is terminated when the value of time exceeds the last table value of the input fore-head pressure trace independent variable TIMEPH(NPH), or when the burn rate table is input, the termination option TST. The steady state interval will then continue with the last table value of the burn rate coefficient dependent variable AKRTAU(NAKRST) or the burn rate coefficient inputs KR(2) and KR(36) (depending on choice of inputs) in the general burning rate equation.

As the burning surface progresses toward the case wall, the anisotropic region is first exposed in the region of sector 8 as shown in Figure 4.4. This results in a non-uniform burning rate along the propellant burning surface during the tail-off interval.

Three burning rates are determined for the burning surface: R_{b8} which is determined from the anisotropic burning distance in sector 8, R_{b7} which is determined from an integration along the anisotropic perimeter of sector 7 between the isotropic and anisotropic burning distances using the anisotropic burn rate coefficient table, and the normal isotropic burn rate, R_b .

Two separate burn distances are defined: the isotropic burn distance in sectors 1 through 7, TAUZ(III), and the anisotropic burn distance in sectors 7 and 8, TAUZT0(III). The anisotropic burn rate becomes progressively less during motor tail-off, resulting in an anisotropic burn distance less than the isotropic burn distance and producing burning that is not normal to the grain surface.

Figures 4.4, 4.5, and 4.6 show the configurations of the anisotropic propellant region that can exist at the case wall for a standard star configuration with an inert sliver. The angles η_2 , η_{22} , and "ANGLE" are used to determine the anisotropic pro-

USE FOR TYPEWRITTEN MATERIAL ONLY

4.3.3

Anisotropic Propellant Burning (Continued)

pellant perimeter lengths, AL_7 and AL_8 , during motor tail-off. Angle η_2 is subtended from the R_5 radius point and locates the intersection of the anisotropic propellant region with the isotropic propellant region. Angle η_{22} is subtended from the motor axis to the same intersection point for η_2 . "ANGLE" is subtended from the motor axis and identifies the point of intersection of the isotropic burn distance vector, swung from R_5 , with the case wall.

Subroutine LPTO contains the geometry calculations to determine the sector perimeter length of the anisotropic propellant for each reference plane during motor tail-off. The anisotropic propellant perimeter length in sector 8 is identified as AL_8 , and in sector 7 is identified as AL_7 . The perimeter length of sector 7 anisotropic propellant (AL_7) is assumed to be a straight line between the points determined by the intersection of the isotropic propellant with the anisotropic propellant and the anisotropic burning distance with the case wall or with sector 8.

The distance burned of the anisotropic propellant in sectors 7 and 8 is computed from the burn rate of the anisotropic propellant in sector 8 (RB_8) in Subroutine SEG SUB. Once sector 8 has burned out, RB_8 is determined from the first table value of the anisotropic burn rate coefficient (the minimum value). The progression of the intersection of the propellant with the case wall is assumed to proceed at the minimum burn rate.

The mass flow generated is determined in subroutine SEG SUB from the perimeter lengths of the anisotropic burning propellant (AL_7 and AL_8), and the isotropic burning propellant (ALP) and their corresponding burn rates as follows:

$$\dot{d}w_7 = NO \rho_f \Delta Z (AL_7 R_{B7} + AL_{7HI} R_{B7HI})$$

$$\dot{d}w_8 = NO \rho_f \Delta Z (AL_8 R_{B8} + AL_{8HI} R_{B8HI})$$

$$\dot{d}w = \left[\frac{\rho_f \Delta Z}{2} (AL_{PHI} - AL_{TOHI}) R_{BHI} + (ALP - AL_{TO}) R_B \right] + \dot{d}w_7 + \dot{d}w_8$$

where

$$AL_{TO} = (AL_7 + AL_8) 2NO$$

4.3.3

Anisotropic Propellant Burning (Continued)

and subscripted "H" values are at the inlet to the mass addition region. The non-subscripted ones represent the outlet to the mass addition region.

USE FOR TYPEWRITTEN MATERIAL ONLY

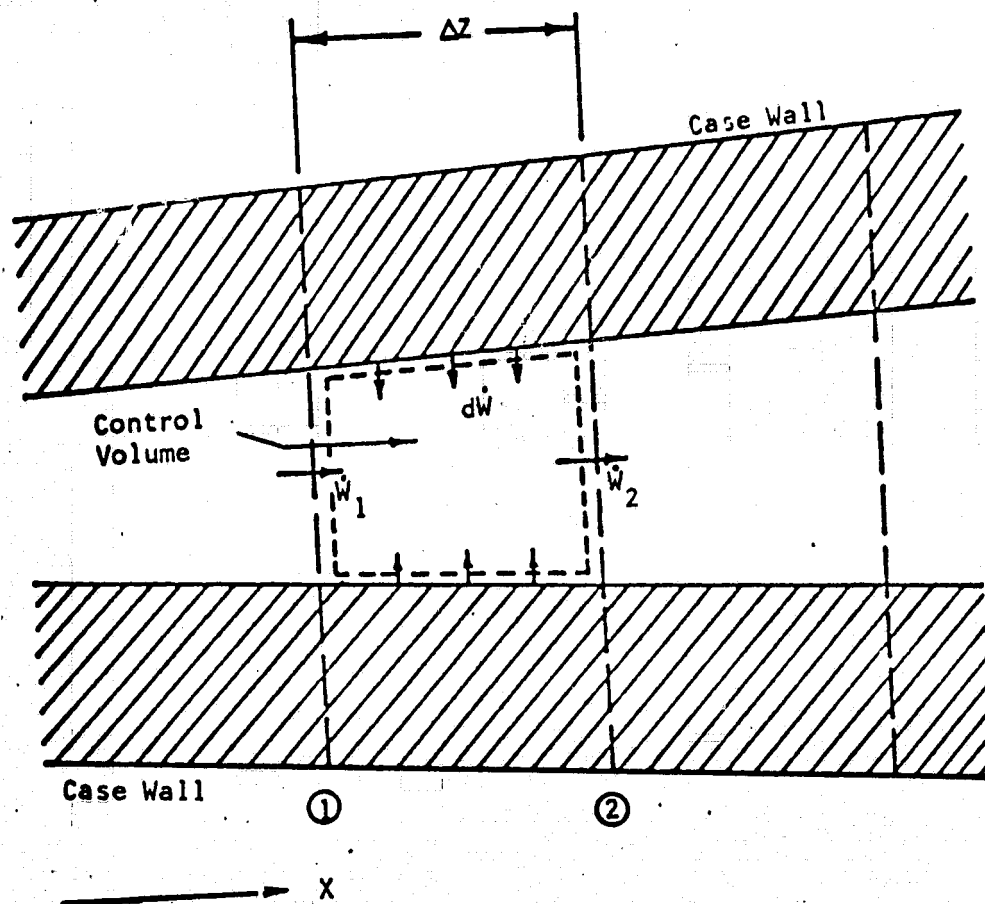
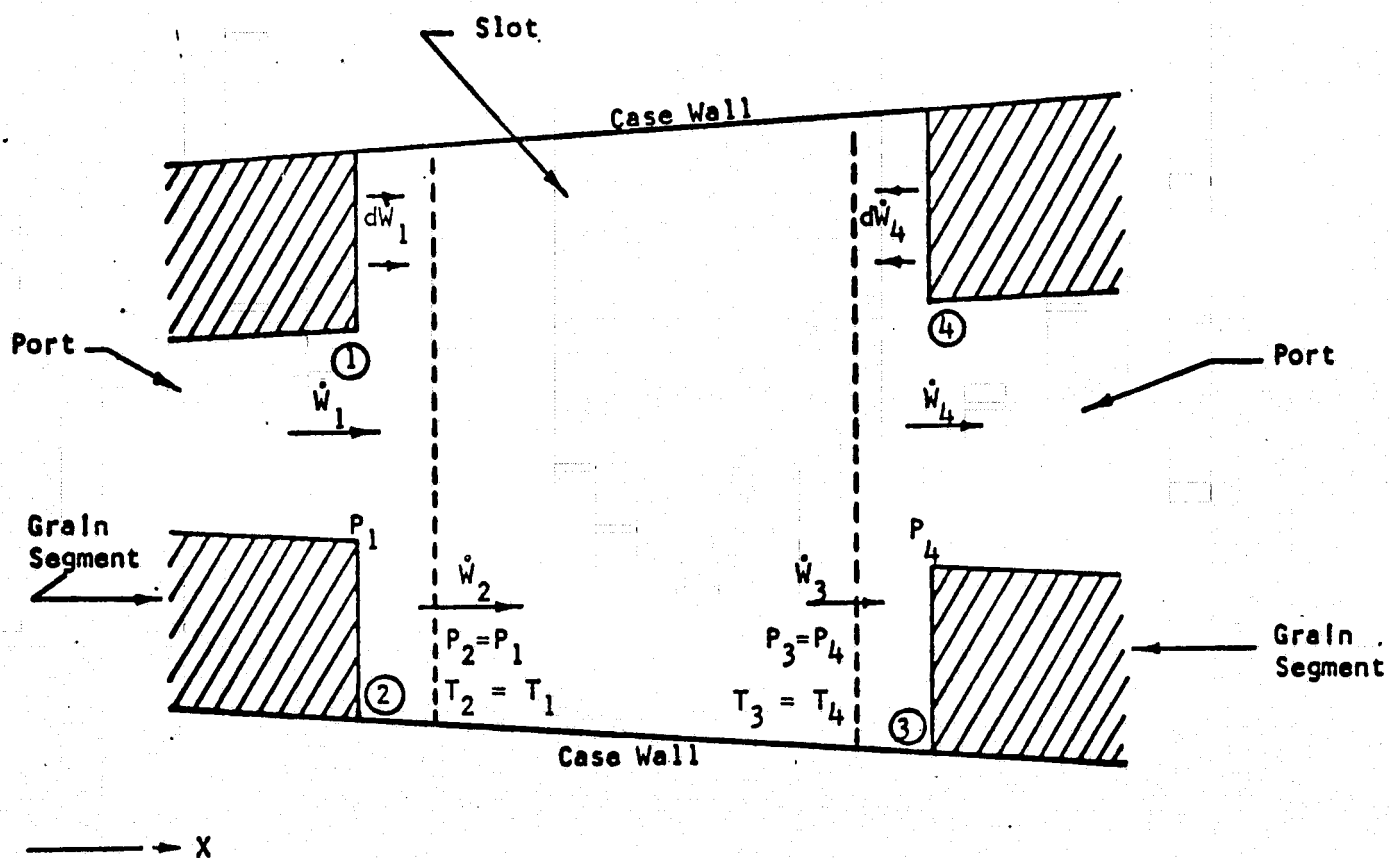


Figure 4.1. Mathematical Model of Mass Addition Region Control Volume



- ① Slot forward Interface
- ② Slot forward Interface mass addition region
- ③ Slot aft Interface mass addition region
- ④ Slot aft Interface

Figure 4.2. Mathematical Model of Slot Between Grain Segments

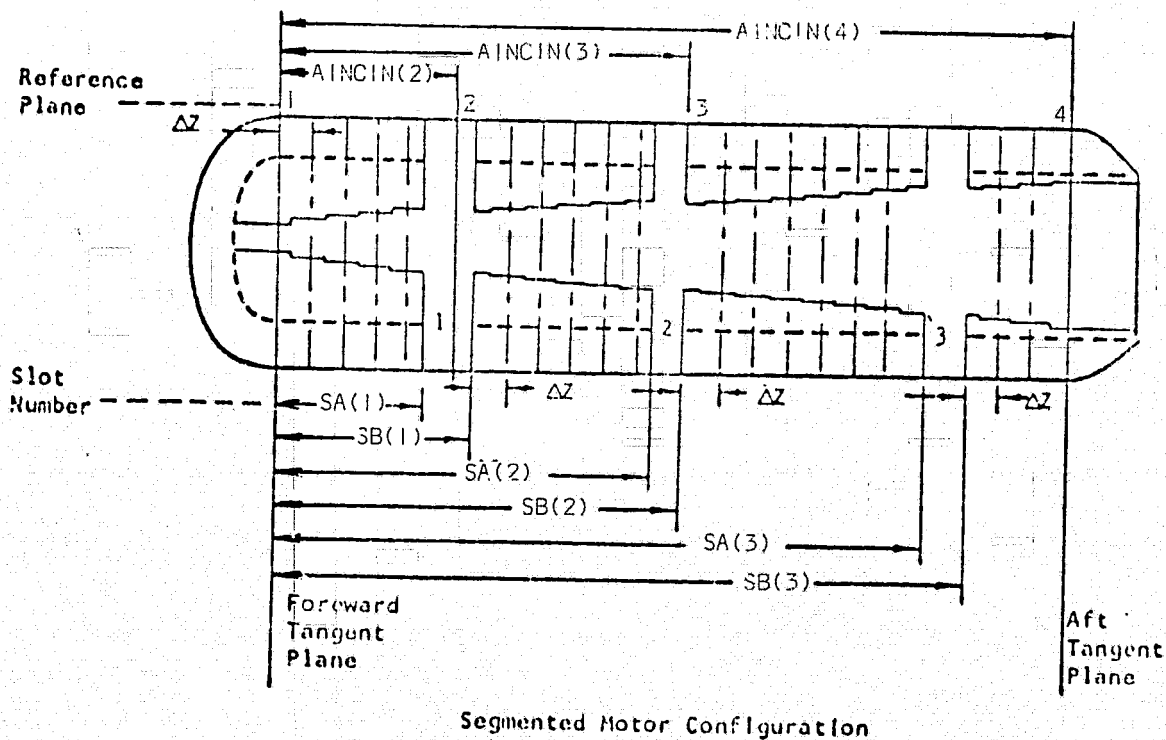
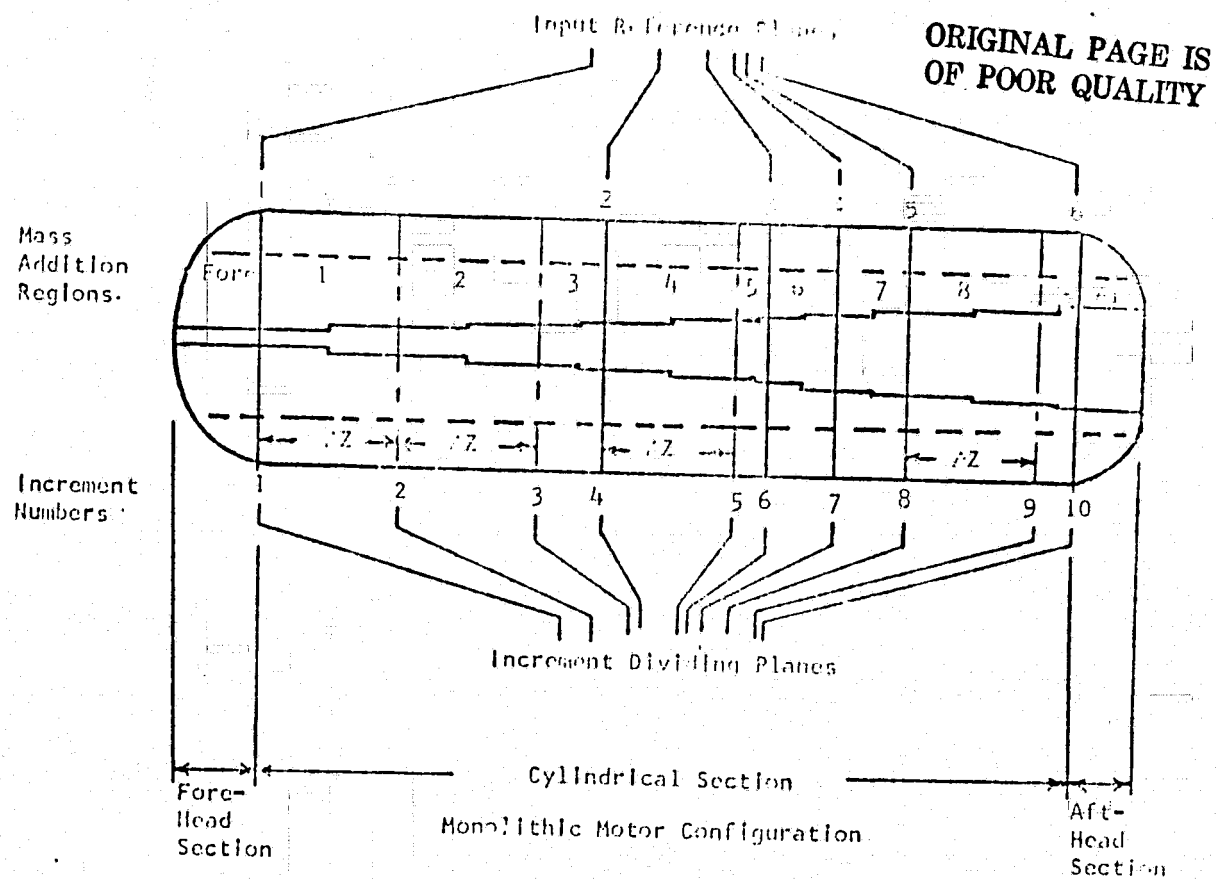


Figure 4.3 Reference Plane and Increment Dividing Plane Identification

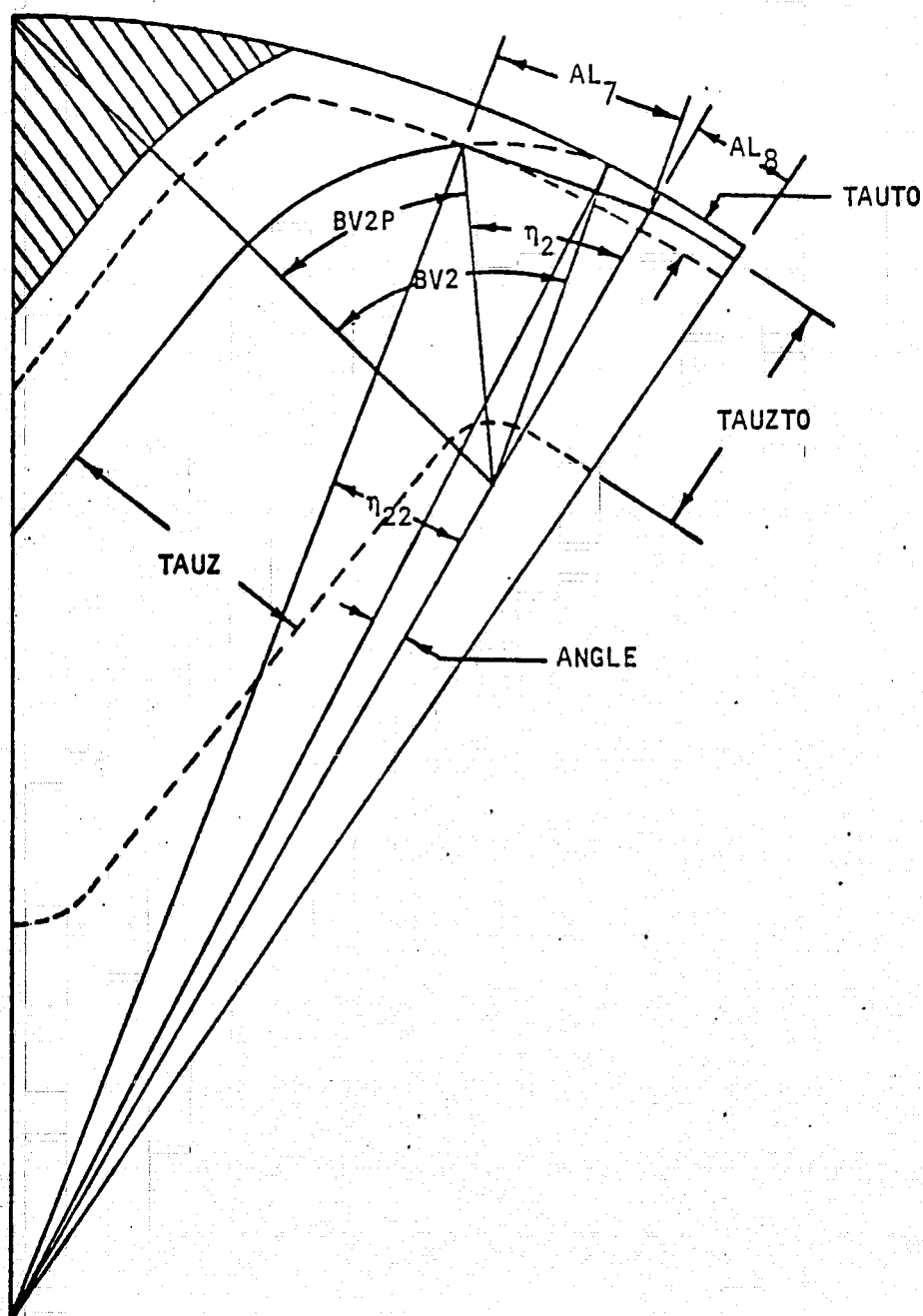


Figure 4.4. Anisotropic Propellant Burning Configuration Near Case Wall Before Web Burnout

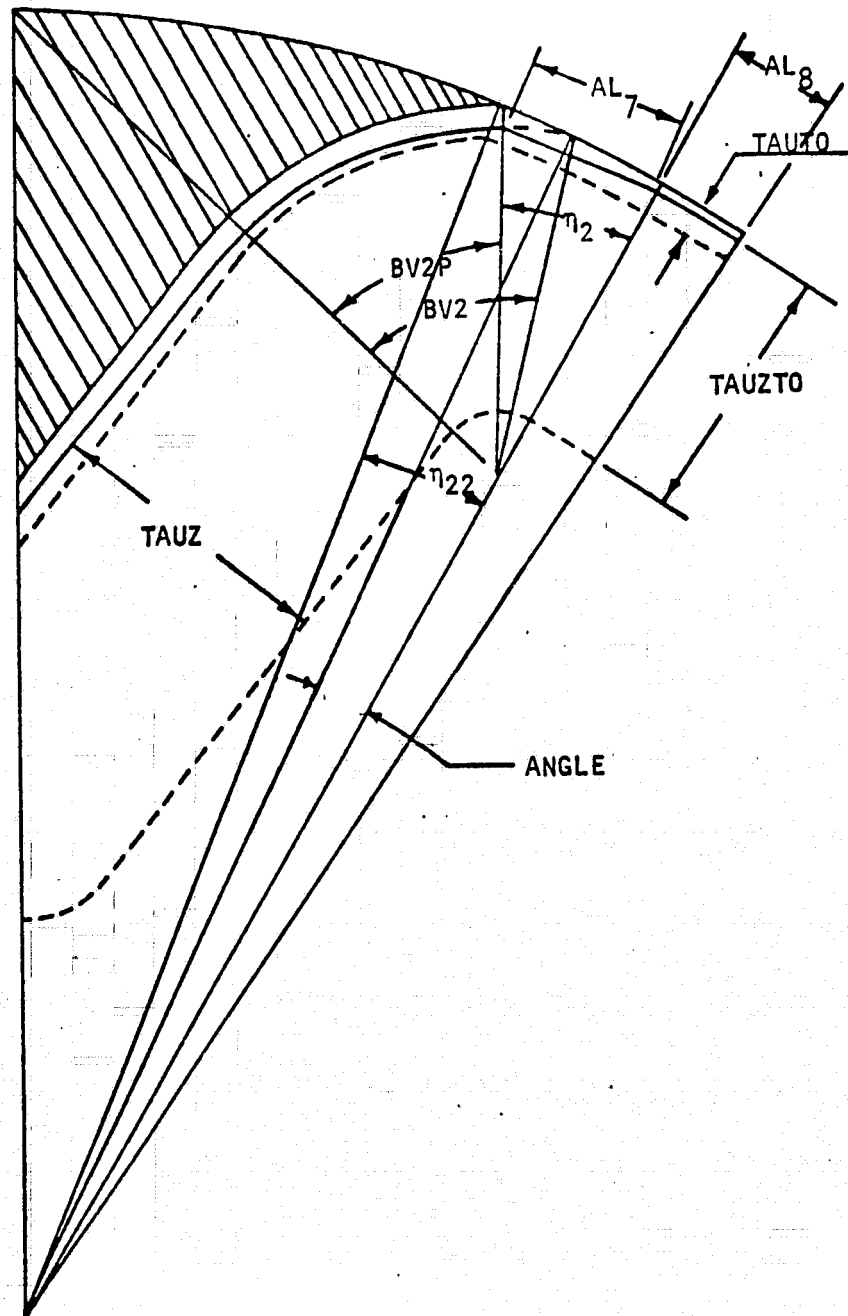


Figure 4.5. Anisotropic Propellant Burning Configuration Near Case Wall and Inert Sliver Before Web Burnout

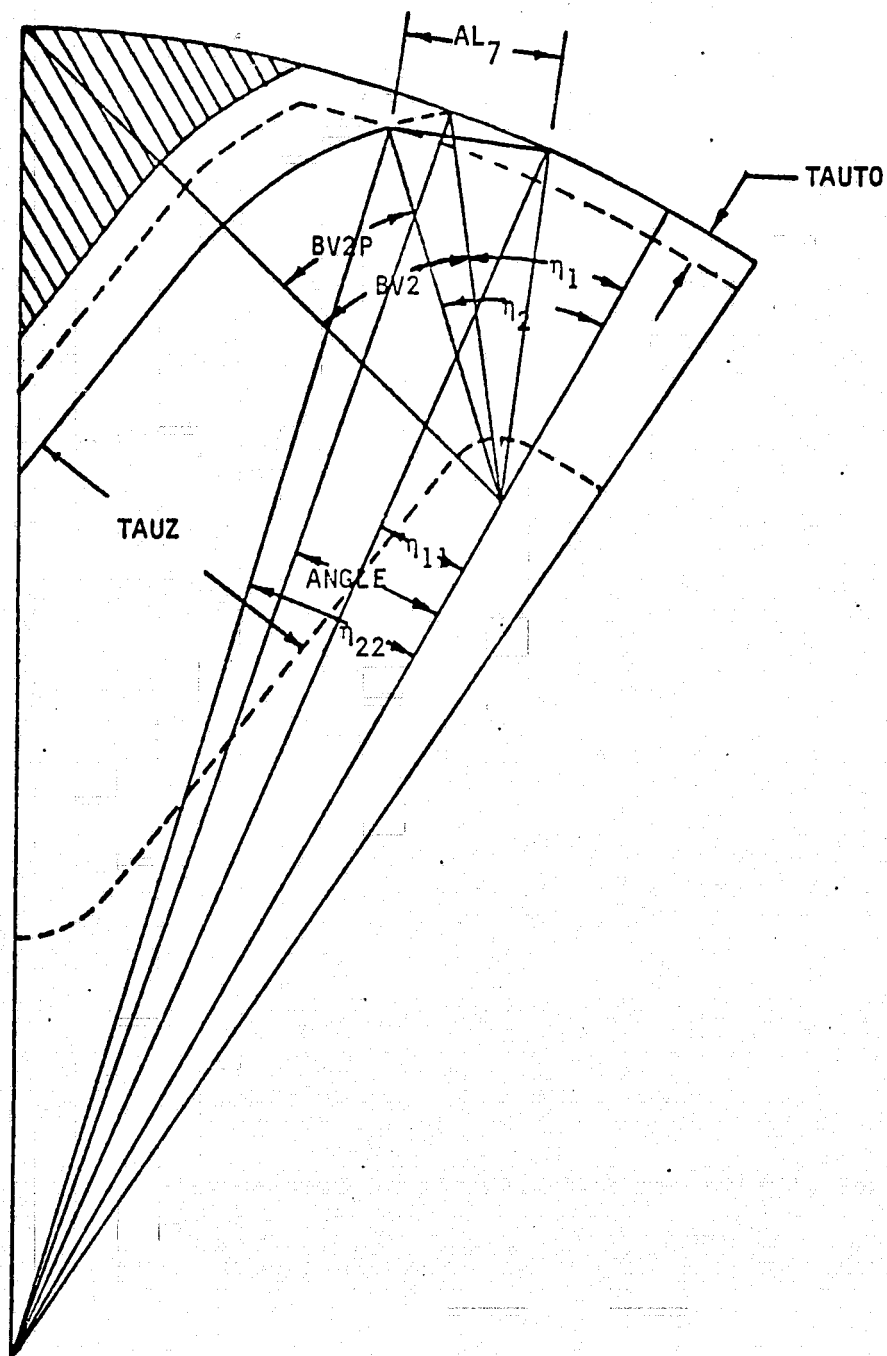


Figure 4.6. Anisotropic Propellant Burning Configuration Near Case Wall After Web Burnout

5.0

GEOMETRICAL DEFINITION OF PROPELLANT GRAIN

For geometrical analysis, the motor is divided longitudinally into three sections: fore-head section, cylindrical section, and aft-head section. The fore-head section may incorporate a head-end with web or it may be defined in a manner similar to the aft-head section which is a straight through grain. The following describes subroutines and equations used to calculate burning surface area, port cross sectional area, propellant volume, center of gravity and moment of inertia for the fore-head, cylindrical, and aft-head sections.

5.1

Grain Cross Section Geometry

Propellant grain cross section options which are programmed for the computer are shown in Figure 5.1. The grain design may vary from the more complicated forked wagon wheel to the standard star, or to the circular port. The slotted-cone grain design is a modification of the standard star. Variations in the input parameters for the forked wagon wheel, which are shown in Figure 5.2, will produce the five basic grain design patterns. The basic parameters required to describe the grain cross section are input for reference planes located at specified distances from the forward tangent plane of the motor, Figure 4.3. The reference planes describing the propellant cross section may be placed at any desired location within the cylindrical section, thus allowing accurate descriptions of the propellant configuration for either monolithic or segmented motors.

5.1.1

General Forked Wagon Wheel

Cross-sectional geometry constants are required to determine the perimeter length and port area of each input reference plane. These basic geometry constants are determined in the first core load. Once these plane constants have been computed and stored for each input reference plane, the perimeter length for all values of distance burned and the initial port area are determined for each reference plane in the second core load and stored in tables for use during the internal ballistic solution in the fourth core load. The reference plane constants for the forked wagon wheel grain configuration are determined by subroutine PLNCNS from the geometry inputs, Figure 5.2. These plane constants are shown in Figures 5.3 and 5.4. A total of 45 constants are generated and stored for each reference plane:

Subroutine PLNCNS Reference Plane Constants

$$R_1 = R_f - \tau_w - L_{S1}$$

$$R_9 = R_f - \tau_w - L_{S2}$$

USE FOR TYPEWRITT. MATERIAL ONLY

5.1.1

General Forked Wagon Wheel (Continued)

$$\theta_1 = \frac{\pi}{N_0}$$

$$\tau_{2\max} = R_2 + L_A \frac{\sin \alpha_{01}}{\cos \alpha_{01}}$$

$$\tau_{4\max} = \tau_{2\max} + L_B \frac{\sin \alpha_{02}}{\cos \alpha_{02}}$$

$$\tau_{5\max} = (\tau_{4\max} + R_4) \frac{\cos \alpha_{02}}{\cos \alpha_{03}} - R_4$$

$$X_{45} = (\tau_{5\max} + R_5) \cos \alpha_{03}$$

$$Y_{45} = R_1 + R_2(1 - \sin \alpha_{01}) + L_A \cos \alpha_{01} + \tau_{2\max}(\sin \alpha_{01} - \sin \alpha_{02}) \\ + L_B \cos \alpha_{02} + R_4(\sin \alpha_{03} - \sin \alpha_{02}) - R_5 \sin \alpha_{03}$$

$$L_C = \left[(R_f - \tau_w - R_5)^2 - (X_{45} \cos \alpha_{03} - Y_{45} \sin \alpha_{03})^2 \right]^{1/2} \\ - X_{45} \sin \alpha_{03} - Y_{45} \cos \alpha_{03}$$

$$X_{03} = (\tau_{2\max} - R_3) \left[\cos \alpha_{01} + \tan \left(\frac{\alpha_{01} - \alpha_{02}}{2} \right) \sin \alpha_{01} \right]$$

$$Y_{03} = R_1 + R_2 + (R_3 - R_2) \sin \alpha_{01} + \left[(\tau_{2\max} - R_3) \tan \frac{\alpha_{01} - \alpha_{02}}{2} \right. \\ \left. + L_A \right] \cos \alpha_{01}$$

$$R_{03} = \left[X_{03}^2 + Y_{03}^2 \right]^{1/2}$$

$$X_{05} = (\tau_{5\max} + R_4) \cos \alpha_{03}$$

5.1.1

General Forked Wagon Wheel (Continued)

$$Y_{05} = Y_{45} + (R_5 - R_4) \sin \alpha_{03}$$

$$R_{05} = \left[X_{05}^2 + Y_{05}^2 \right]^{1/2}$$

$$X_{07} = X_{45} + L_C \sin \alpha_{03}$$

$$Y_{07} = Y_{45} + L_C \cos \alpha_{03}$$

$$R_{07} = \left[X_{07}^2 + Y_{07}^2 \right]^{1/2}$$

$$\tau_{6\max} = \tau_{5\max} + L_C \frac{\sin \alpha_{03}}{\cos \alpha_{03}}$$

$$\tau_{7\max} = \left[X_{07}^2 + (R_f - Y_{07})^2 \right]^{1/2} - R_5$$

$$\tau_{12\max} = R_8 + L_E \frac{\sin \alpha_{05}}{\cos \alpha_{05}}$$

$$X_{76} = (\tau_{12\max} + R_6) \cos \alpha_{04}$$

$$Y_{76} = R_9 + R_8(1 - \sin \alpha_{05}) + L_E \cos \alpha_{05} + \tau_{12\max}(\sin \alpha_{05} - \sin \alpha_{04}) - R_6 \sin \alpha_{04}$$

$$L_D = \left[(R_f - \tau_w - R_6)^2 - (X_{76} \cos \alpha_{04} - Y_{76} \sin \alpha_{04})^2 \right]^{1/2} - X_{76} \sin \alpha_{04} - Y_{76} \cos \alpha_{04}$$

$$X_{09} = X_{76} + L_D \sin \alpha_{04}$$

$$Y_{09} = Y_{76} + L_D \cos \alpha_{04}$$

USE FOR TYPEWRITTEN MATERIAL ONLY

5.1.1

General Forked Wagon Wheel (Continued)

$$R_{09} = [X_{09}^2 + Y_{09}^2]^{1/2}$$

$$X_{011} = (\tau_{12\max} - R_7) \left[\tan \frac{\alpha_{05} - \alpha_{04}}{2} \sin \alpha_{05} + \cos \alpha_{05} \right]$$

$$Y_{011} = R_9 + R_8 + (R_7 - R_8) \sin \alpha_{05} + [(\tau_{12\max} - R_7) \tan \left(\frac{\alpha_{05} - \alpha_{04}}{2} \right) + L_E] \cos \alpha_{05}$$

$$R_{011} = [X_{011}^2 + Y_{011}^2]^{1/2}$$

$$\tau_{10\max} = \tau_{12\max} + L_D \frac{\sin \alpha_{04}}{\cos \alpha_{04}}$$

$$\tau_{9\max} = [X_{09}^2 + (R_f - Y_{09})^2]^{1/2} - R_6$$

$$\theta_2 = \arccos \left(\frac{Y_{07}}{R_{07}} \right)$$

$$\theta_3 = \arccos \left(\frac{Y_{09}}{R_{09}} \right)$$

$$\theta_4 = \theta_1 - \theta_2 - \theta_3$$

$$\beta_{71\max} = \arccos \left[\frac{(\tau_{7\max} + R_5)^2 - (R_f - Y_{07})^2}{(\tau_{7\max} + R_5)^2} \right]^{1/2} / (\tau_{7\max} + R_5) - \alpha_{03}$$

$$\beta_{72\max} = \frac{\pi}{2} + \theta_2 - \beta_{71\max} - \alpha_{03}$$

$$\beta_{91\max} = \arccos \left[\frac{(\tau_{9\max} + R_6)^2 - (R_f - Y_{09})^2}{(\tau_{9\max} + R_6)^2} \right]^{1/2} / (\tau_{9\max} + R_6)$$

5.1.1

General Forked Wagon Wheel (Continued)

$$\beta_{92\max} = \frac{\pi}{2} + \theta_3 - \beta_{91\max} - \alpha_{04}$$

$$\tau_{\max} = \tau_{7\max}, \text{ or } \tau_{9\max}, \text{ whichever is larger.}$$

If θ_0 is input, determine τ_{\max} for slotted-cone geometry as follows (Figure 5.7):

$$\angle AJB = \arcsin \left[\frac{(R_f - R_5 - \tau_w) \sin(\theta_2)}{T_{6M} + R_5} \right]$$

$$\angle ABC = \pi - \theta_2 - \angle AJB$$

$$\angle ACB = \arcsin \left[\frac{(R_f - R_5 - \tau_w) \sin(\angle ABC)}{R_f} \right]$$

$$\angle BAC = \pi - \angle ABC - \angle ACB$$

$$\angle GFE = \frac{\pi}{2} - \alpha_{01}$$

$$\angle AFE = \frac{\pi}{2} + \alpha_{01}$$

$$\angle AEF = \arcsin \left[\frac{(R_1 + R_2) \sin(\angle AFE)}{R_f} \right]$$

$$\angle GAE = \pi - \angle AFE - \angle AEF$$

$$\text{If } \theta_0 \geq \angle GAE, \tau_{\max} = R_f - R_1$$

$$\text{If } \theta_0 < \angle GAE,$$

$$BH = [R_f^2 + (R_f - R_5 - \tau_w)^2 - 2R_f(R_f - R_5 - \tau_w) \cos(\theta_0 + \theta_2)]^{\frac{1}{2}}$$

$$\text{If } (\theta_0 + \theta_2) \geq \angle BAC,$$

$$\angle AHB = \arcsin \left[\frac{(R_f - R_5 - \tau_w) \sin(\theta_0 + \theta_2)}{BH} \right]$$

$$\angle ABH = \pi - \angle AHB - \theta_0 - \theta_2$$

$$\angle HBC = \angle ABC - \angle ABH$$

$$BH' = BH \cos(\angle HBC)$$

5.1.1

General Forked Wagon Wheel (Continued)

$$\tau_{\max} = (R_f - R_1) \text{ or } (BH' - R_5), \text{ whichever is smaller.}$$

$$\text{If } (\theta_0 + \theta_2) < \angle BAC,$$

$$\tau_{\max} = (R_f - R_1) \text{ or } (BH - R_5), \text{ whichever is smaller.}$$

The calculated plane constants appear on the printout and allow the user a means of checking if the proper grain design is being analyzed. Although the program is designed to solve the general configuration shown in Figure 5.2, there are certain variations of this configuration that exceed the mathematical limits of the analysis. To obtain a program solution, all of the following conditions must exist for each reference plane used:

R_1	must be greater than or equal to zero
R_3	must be less than or equal to $\tau_{2\max}$
R_7	must be less than or equal to $\tau_{12\max}$
R_9	must be greater than or equal to zero
L_C	must be greater than or equal to zero
L_D	must be greater than or equal to zero
β_{71M}	must be greater than or equal to zero
β_{91M}	must be greater than or equal to zero
α_{01}	must be less than 90°
α_{02}	must be greater than or equal to zero
α_{03}	must be less than 90°
α_{04}	must be greater than or equal to zero
α_{05}	must be less than 90°
θ_4	must be greater than or equal to zero

If any of these restrictions are exceeded for any of the input reference planes, the program will automatically stop and print the reference plane dimensions along with a statement of the exceeded restriction. Although the program is self-checking for these mathematically invalid configurations, there are some physically invalid configurations for which there is no such check.

Special attention should be given to the manner in which input lengths L_A , L_B , and L_E are defined. The length L_A is measured along a common tangent to the arcs generated by radii R_2 and R_3 (Figure 5.2). One of the points that defines this length is the point of tangency with the arc generated by the radius R_2 . How-

5.1.1 General Forked Wagon Wheel (Continued)

ever, the other point that defines length L_A is not the point of tangency with the arc generated by radius R_3 . This end limit of length L_A is determined by bisecting the included angle between lengths L_A and L_B , and extending this bisector until it intersects the axis of symmetry of the grain sector. From this intersection point, a line perpendicular to L_A is drawn. The intersection of this perpendicular and the line L_A is the point which defines the end of length L_A . Line lengths L_B and L_E are defined in a similar manner.

The perimeter length and initial port area of the grain cross section for each reference plane are determined in the second core load for incremental distances burned before and after web burn-out specified by input. The geometry plane constants of each reference plane are moved from storage into working locations in subroutine LPDAPS and the perimeter length and initial port area are determined for each sector of the cross section (shown in Figures 5.2, 5.3, and 5.4) at the specified DTAU increments by subroutine AFPSUB.

Subroutine LPDAPS

Subroutine LPDAPS sets up the parameters required to determine the perimeter length, L_p , and the cross-sectional propellant area, A_{FF} , for both the primary and secondary propellant tips of the forked wagon wheel. Initially, the parameters required to determine the secondary propellant tip are set and subroutine AFPSUB is used to evaluate the perimeter length and the area of propellant for each sector, and then the parameters for the primary propellant tip are set. Figure 5.5 shows the boundaries of the sectors for the forked wagon wheel grain.

The perimeter length and propellant area of sector 8 are then determined and summed with the values determined from sectors 1-7 and 9-13. The total perimeter of the grain configuration is obtained by multiplying the sum by $2N_0$. The port area is obtained by summing the cross sectional area of the propellant sectors, multiplying by N_0 , and subtracting the result from the area of a circle of radius R_f .

Subroutine AFPSUB

Subroutine AFPSUB determines the port perimeter length of one-

USE FOR TYPEWRITTEN MATERIAL ONLY

5.1.1

General Forked Wagon Wheel (Continued)

half of a symmetrical section and propellant cross sectional area of a symmetrical section of all sectors except sector 8. The plane constants required for each sector are set by subroutines MNCHN2 and LPDAPS.

The perimeter and area of the cross section for all grain configurations are calculated from basic trigonometric formulas. Required angles and line lengths are computed from known dimensions using the law of sines and the law of cosines. Propellant cross sectional area is determined by adding and subtracting areas of circular sectors and triangles. All configurations contain only straight lines and circular arcs.

Initially, the constants to determine the initial area and perimeter length for sectors 9 through 13, Figure 5.5, are set by subroutine LPDAPS, and then the sectors 1 through 7 are set such that $L_{13} = L_1$, $L_{12} = L_2$, $L_{11} = L_3$, $L_{10} = L_4$, and $L_9 = L_7$. The sector perimeter length, L , and initial propellant cross sectional areas, AFP, are determined as follows:

$$1. \quad \text{If } \tau < R_2,$$

$$L_1 = (R_2 - \tau) \left(\frac{\pi}{2} - \alpha_{01} \right)$$

$$AFP_1 = (R_2 - \tau) L_1$$

$$2. \quad \text{If } \tau \geq R_2,$$

$$L_1 = 0$$

$$AFP_1 = 0$$

$$3. \quad \text{If } \tau < R_3,$$

$$\Delta L_3 = (\tau_{2\max} - R_3) \tan \left(\frac{\alpha_{01} - \alpha_{02}}{2} \right)$$

$$AFD = (R_3 + \tau_{2\max} - 2\tau) \Delta L_3$$

USE FOR TYPEWRITTEN MATERIAL ONLY

5.1.1

General Forked Wagon Wheel (Continued)

4. If
- $\tau \geq R_3$
- and if
- $\tau < \tau_{2\max}$

$$\Delta L_3 = (\tau_{2\max} - \tau) \tan \left(\frac{\alpha_{01} - \alpha_{02}}{2} \right)$$

$$AFD = (\tau_{2\max} - \tau) \Delta L_3$$

5. If
- $\tau \geq R_3$
- and if
- $\tau \geq \tau_{2\max}$

$$L_2 = 0$$

$$\Delta L_3 = 0$$

$$AFD = 0$$

6. If
- $\tau \leq R_2$
- ,

$$L_2 = L_A + \Delta L_3$$

$$AFP_2 = (R_2 - 2\tau + \tau_{2\max}) L_A + AFD$$

7. If
- $\tau > R_2$
- ,

$$L_2 = L_A \left[\frac{(\tau_{2\max} - \tau)}{(\tau_{2\max} - R_2)} \right] + \Delta L_3$$

$$AFP_2 = (\tau_{2\max} - \tau) (L_2 + \Delta L_3) + AFD$$

8. If
- $\tau < R_3$
- ,

$$L_3 = (R_3 - \tau) (\alpha_{01} - \alpha_{02})$$

$$AFP_3 = L_3 (R_3 - \tau)$$

9. If
- $\tau \geq R_3$
- ,

$$L_3 = 0$$

$$AFP_3 = 0$$

USE FOR TYPEWRITTEN MATERIAL ONLY

5.1.1

General Forked Wagon Wheel (Continued)

10. If $\tau < \tau_{2\max}$

$$L_4 = L_B + \Delta L_3$$

$$AFP_4 = (\tau_{2\max} - 2\tau + \tau_{4\max}) L_B + AFD$$

11. If $\tau_{2\max} \leq \tau < \tau_{4\max}$

$$L_4 = (\tau_{4\max} - \tau) L_B / (\tau_{4\max} - \tau_{2\max})$$

$$AFP_4 = (\tau_{4\max} - \tau) L_4$$

12. If $\tau_{2\max} \leq \tau \geq \tau_{4\max}$

$$L_4 = 0$$

$$AFP_4 = 0$$

13. If $\tau < \tau_{4\max}$

$$L_5 = (R_4 + \tau) (\alpha_{03} - \alpha_{02})$$

$$AFP_5 = (R_4 + \tau_{5\max}) (R_4 + \tau_{4\max}) \sin(\alpha_{03} - \alpha_{02}) - L_5 (R_4 + \tau)$$

14. If $\tau_{4\max} \leq \tau < \tau_{5\max}$

$$L_5 = (R_4 + \tau) \alpha_{03} - \arccos \left[\frac{(R_4 + \tau_{4\max}) \cos(\alpha_{02})}{(R_4 + \tau)} \right]$$

$$AFP_5 = (R_4 + \tau_{5\max}) (R_4 + \tau) \sin \left[\frac{L_5}{(R_4 + \tau)} \right] - (R_4 + \tau) L_5$$

15. If $\tau_{4\max} \leq \tau \geq \tau_{5\max}$

$$L_5 = 0$$

$$AFP_5 = 0$$

USE FOR TYPEWRITTEN MATERIAL ONLY

5.1.1

General Forked Wagon Wheel (Continued)

16. If $\tau \leq \tau_{5\max}$,

$$L_6 = L_C$$

$$AFP_6 = (\tau_{5\max} - 2\tau + \tau_{6\max}) L_C$$

17. If $\tau_{5\max} < \tau < \tau_{6\max}$ L_C

$$L_6 = L_C (\tau_{6\max} - \tau) / (\tau_{6\max} - \tau_{5\max})$$

$$AFP_6 = (\tau_{6\max} - \tau) L_6$$

18. If $\tau_{5\max} < \tau \geq \tau_{6\max}$,

$$L_6 = 0$$

$$AFP_6 = 0$$

19. If $\tau \leq \tau_{6\max}$,

$$\beta_{71} = \beta_{71\max}$$

$$AFP = (R_5 + \tau_{7\max})(R_5 + \tau_{6\max}) \sin(\beta_{71}) - \beta_{71}(R_5 + \tau)^2$$

20. If $\tau_{6\max} < \tau < \tau_{7\max}$,

$$\beta_{71} = \beta_{71\max} + \alpha_{03} - \arccos \left(\frac{x_{07}}{(\tau + R_5)} \right)$$

$$AFP = (R_5 + \tau_{7\max})(R_5 + \tau) \sin(\beta_{71}) - \beta_{71}(R_5 + \tau)^2$$

21. If $TSLVR > 0$,

$$\beta_{VX} = \arcsin \left[\frac{R_7 \sin(\theta_2)}{R_5 + \tau_{7\max}} \right] - \arcsin \left[\frac{R_7 \sin(\theta_2 - \theta_{SLVR})}{(R_5 + \tau_{SLVR})} \right] - \theta_{SLVR}$$

$$\text{Temp} = \tau_w^2 + (R_5 + \tau_{7\max})^2 - 2 \tau_w (R_5 + \tau_{7\max})$$

$$\cos \left[\arcsin \left(\frac{R_7 \sin \theta_2}{R_5 + \tau_{7\max}} \right) \right]$$

5.1.1 General Forked Wagon Wheel (Continued)

$$\text{Temp}_a = \arcsin \left[\frac{R_7 \sin \theta_2}{R_5 + \tau_{7\max}} \right]$$

$$\text{Temp}_b = \pi - \arcsin \left[\frac{R_7 \sin \theta_2}{\text{Temp}} \right]$$

$$\text{Temp}_c = \arcsin \left[\frac{\text{Temp} \sin(\text{Temp}_b)}{(T_{\text{SLVR}} + R_5)} \right]$$

$$\beta_{\text{VXX}} = \text{Temp}_c - \text{Temp}_a$$

$$\begin{aligned} A_{\text{SLVR}} = & \theta_2 R_f^2 - (R_f - \tau_w) R_7 \sin \theta_2 - (\theta_2 - \theta_{\text{SLV}}) R_f^2 \\ & + (T_{\text{SLVR}} + R_5) R_7 \sin \left\{ \pi - \theta_2 + \theta_{\text{SLV}} \right. \\ & \left. - \arcsin \left[\frac{R_7 \sin(\theta_2 - \theta_{\text{SLV}})}{(T_{\text{SLVR}} + R_5)} \right] \right\} \\ & - \text{Temp} (T_{\text{SLVR}} + R_5) \sin(\pi - \text{Temp}_c - \text{Temp}_b) \\ & - (\beta_{\text{VXX}} + \beta_{\text{VX}}) (T_{\text{SLVR}} + R_5)^2 \end{aligned}$$

22. If $\tau \leq \tau_w$,

$$\beta_{72} = \beta_{72\max}$$

$$\text{AFP}_7 = \text{AFP} + \theta_2 R_f^2 - R_f R_7 \sin \theta_2 - \beta_{72} (R_5 + \tau)^2$$

$$L_7 = (\beta_{71} + \beta_{72}) (R_5 + \tau)$$

23. If $\tau > \tau_w$,

$$\begin{aligned} \beta_{72} = & \arccos \left\{ \frac{(\tau + R_5)^2 + (R_f - \tau_w - R_5)^2 - R_f^2}{2(\tau + R_5)(R_f - \tau_w - R_5)} \right\} \\ & - \beta_{71\max} - \alpha_{03} - \frac{\pi}{2} + \theta_2 \end{aligned}$$

5.1.1

General Forked Wagon Wheel (Continued)

$$AFP_7 = AFP + \theta_2 R_f^2 - R_f R_7 \sin \theta_2 + R_7 (R_5 + \tau) \sin(\beta_{72\max} - \beta_{72})$$

$$- R_f^2 \arccos \left\{ 1 - \left(\frac{(R_5 + \tau)(\sin(\beta_{72\max} - \beta_{72}))^2}{R_f} \right)^{1/2} \right\}$$

$$- \beta_{72} (R_5 + \tau)^2$$

$$L_7 = (\beta_{71} + \beta_{72}) (R_5 + \tau)$$

24. If $\tau < \tau_{SLVR}$,

$$L = L_1 + L_2 + L_3 + L_4 + L_5 + L_6 + L_7$$

$$AFP = AFP_1 + AFP_2 + AFP_3 + AFP_4 + AFP_5 + AFP_6 + AFP_7 - ASLVR$$

25. If $\tau \geq \tau_{SLVR}$,

$$L = 0$$

$$AFP = 0$$

5.1.2

Slotted-Cone

The slotted-cone configuration is an addition to the standard star in which a segment of propellant, represented by angle θ_0 , has been inserted as shown in Figure 5.6. The insert of propellant results in additions to the perimeter and area calculations for a standard star which requires the determination of three basic angles, QAG, RAG, and SAG, shown in Figures 5.9, 5.10, and 5.11, respectively. Nine points, A, B, C, C', E, F, G, J, and H are defined in Figure 5.7, and are used in the following analysis to determine line lengths and angles. Four moving points, P, Q, R, and S on the burning perimeter are identified in Figures 5.8 through 5.13.

Point A lies on the motor axis (Figure 5.7). Point B coincides with the center of the input radius R_5 and point F coincides with the center of the input radius R_2 . Point G is located on the case

5.1.2 Slotted-Cone (Continued)

wall at the point where a line through points A and F intersect the case. Points C and E are located on the case wall where a line perpendicular to side L_C at the intersection of side L_C with radius R_5 intersects the case and where a line perpendicular to side L_C at the intersection of side L_C with radius R_2 intersects the case. Point H lies on the case wall at the angle θ_0 from point G. Point C' defines the point where side L_C of the standard star disappears as a result of the progression of the burning surface of the slotted cone insert.

Perimeter length QR, Figure 5.9, defines the addition to side L_C of the standard star, resulting from the addition of the central angle θ_0 of the slotted-cone. Perimeter length PQ defines the length of the arc subtended from the radius, $(R_1 + \tau)$, and intersecting with length QR. When the burning distance, τ , is less than or equal to the input radius R_2 , only the perimeter length PQ is present. When the burning distance is greater than R_2 , but less than or equal to the geometry constant T_{6max} , as shown in Figure 5.9, both perimeter lengths PQ and QR are present.

Perimeter length RS defines the addition to the arc subtended from the point of the input radius R_5 (geometry point B). Whenever T_{V7}' is greater than T_{7max} , and T_{7max} is greater than T_{6max} (Figure 5.10), and τ is greater than T_{6max} but less than or equal to T_{7max} , geometry point S lies on the line AG and perimeter lengths PQ, QR, and RS are present.

Whenever τ is greater than T_{7max} but less than T_{V7}' , perimeter length RS is defined as in Figure 5.11.

Whenever τ is greater than T_{V7}' , as shown in Figures 5.12 and 5.13, geometry point R and perimeter length QR do not exist and only perimeter lengths PQ and QS remain.

The angle θ_0 is allowed to vary on input between 0° and 90° . Should θ_0 be less than the angle QAG shown in Figure 5.9, then the perimeter length PQ does not exist and point Q then lies on the line AH. Should θ_0 be less than the angles QAG and RAG, then the perimeter lengths PQ and QR and the points P and Q do not

USE FOR TYPEWRITTEN MATERIAL ONLY

5.1.2

Slotted-Cone (Continued)

exist and the point R lies on the line AH. When angle SAG exceeds θ_0 , complete burnout has occurred.

The following are the port perimeter and propellant cross sectional area equations for this addition to the standard star configuration. Figures 5.8 through 5.13 show the progression of the burning surface and the geometric figures required for each equation.

1. If $\tau \leq R_2$ (Figure 5.8),

$$\text{Perimeter} = \theta_0 (R_1 + \tau)$$

$$\text{Area} = \theta_0 \left[R_f^2 - (R_1 + \tau)^2 \right]$$

2. If $\tau > R_2$ (Figure 5.9),

$$\angle AJB = \arcsin \left[\frac{(R_f - R_5 - \tau_w)}{T_{6\max} + R_5} \sin \theta_2 \right]$$

$$\angle ABC = \pi - \theta_2 - \angle AJB$$

$$\angle ACB = \arcsin \left[\frac{(R_f - R_5 - \tau_w)}{R_f} \sin \angle ABC \right]$$

$$\angle BAC = \pi - \angle ABC - \angle ACB$$

$$BC = R_f \frac{\sin \angle BAC}{\sin \angle ABC}$$

$$T'_{V6} = BC - R_5$$

$$T'_{V7} = \frac{(R_f - R_5 - \tau_w)^2 - R_1^2 + R_5^2 - 2(R_f - R_5 - \tau_w)R_5 \cos \angle ABC}{2R_1 - 2R_5 + 2(R_f - R_5 - \tau_w) \cos \angle ABC}$$

$$\text{Temp}_{76} = \text{minimum of } T'_{V7} \text{ and } T'_{V6}$$

5.1.2

Slotted-Cone (Continued)

3. If $R_2 < T_{67}$, and if $\tau \leq T_{6\max}$ (Figure 5.9),

$$FR = \frac{(\tau - R_2)}{\sin \alpha_{01}}$$

$$AR = R_1 + R_2 + FR$$

$$\angle RQA = \pi - \arcsin \left[\frac{AR \sin \alpha_{01}}{R_1 + \tau} \right]$$

$$\angle QAR = \pi - \angle RQA - \alpha_{01}$$

$$QR = (R_1 + \tau) \frac{\sin \angle QAR}{\sin \alpha_{01}}$$

$$\angle QAG = \arcsin \left[\frac{QR \sin \alpha_{01}}{(R_1 + \tau)} \right]$$

a. and $\angle QAG \geq \theta_0$,

$$\text{Perimeter} = \frac{AR \sin \theta_0}{\sin(\pi - \alpha_{01} - \theta_0)}$$

$$\text{Area} = \theta_0 R_f^2 - (\text{Perimeter}) AR \sin \alpha_{01}$$

b. Or $\angle QAG < \theta_0$,

$$PQ = (R_1 + \tau)(\theta_0 - \angle QAG)$$

$$\text{Perimeter} = PQ + QR$$

$$\text{Area} = \theta_0 R_f^2 - (\theta_0 - \angle QAG)(R_1 + \tau)^2 - QR AR \sin \alpha_{01}$$

USE FOR TYPEWRITTEN MATERIAL ONLY

5.1.2

Slotted-Cone (Continued)

4. If $R_2 < \tau < \text{Temp}_{76}$ and if $T_{6\max} < \tau \leq T_{7\max}$ (Figure 5.10),

$$\angle ASB = \arcsin \left[\frac{(R_f - R_5 - \tau_w) \sin \theta_2}{(R_5 + \tau)} \right]$$

$$\angle SBA = \pi - \theta_2 - \angle ASB$$

$$\text{Area}_1 = \theta_0 R_f^2 + (R_f - R_5 - \tau_w)(R_5 + \tau) \sin \angle SBA$$

a. And $\angle RAG \geq \theta_0$

$$\angle BRA = \arcsin \left[\frac{(R_f - R_5 - \tau_w) \sin(\theta_2 + \theta_0)}{(R_5 + \tau)} \right]$$

$$\angle RBA = \pi - \angle BRA - \theta_2 - \theta_0$$

$$\angle RBS = \angle SBA - \angle RBA$$

$$RS = \angle RBS(R_5 + \tau)$$

$$\text{Perimeter} = RS$$

$$\text{Area} = \text{Area}_1 - (R_f - R_5 - \tau_w)(R_5 + \tau) \sin \angle RBA - \angle RBS(R_5 + \tau)^2$$

b. Or $\angle RAG < \theta_0$

$$\text{Area}_2 = \text{Area}_1 - \angle RBS(R_5 + \tau)^2 - (R_f - R_5 - \tau_w)(R_5 + \tau) \sin \angle ABC$$

(1) And If $\angle QAG \geq \theta_0$

$$\angle QAR = \theta_0 - \angle RAG$$

$$\angle AQR = \pi/2 + \angle ARB - \angle QAR$$

$$QR = \frac{AR \sin \angle QAR}{\sin \angle AQR}$$

$$\text{Perimeter} = QR + RS$$

USE FOR TYPEWRITTEN MATERIAL ONLY

5.1.2 Slotted-Cone (Continued)

$$\text{Area} = \text{Area}_2 - QR AR \sin(\pi/2 - \angle ARB)$$

$$(2) \text{ or if } \angle QAG < \theta_0$$

$$\angle PAQ = \theta_0 - \angle QAG$$

$$PQ = \angle PAQ (R_1 + \tau)$$

$$QR = \frac{(R_1 + \tau) \sin \angle RAQ}{\sin(\pi/2 - \angle ARB)}$$

$$\text{Perimeter} = PQ + QR + RS$$

$$\text{Area} = \text{Area}_2 - QR AR \sin(\pi/2 - \angle ARB) - \angle PAQ (R_1 + \tau)^2$$

5. If $R_2 < \tau < \text{Temp}_{76}$, and if $T_{6\max} < \tau < T_{7\max}$ (Figure 5.11),

$$\angle SAB = \arccos \left[\frac{R_f^2 + (R_f - R_5 - \tau_w)^2 - (R_5 + \tau)^2}{2 R_f (R_f - R_5 - \tau_w)} \right]$$

$$\angle SAG = \angle SAB - \theta_2$$

$$AR = \left[(R_5 + \tau)^2 + (R_f - R_5 - \tau_w)^2 - 2(R_5 + \tau)(R_f - R_5 - \tau_w) \cos \angle ABC \right]^{1/2}$$

$$\angle RAB = \arcsin \left[\frac{(R_5 + \tau) \sin \angle ABC}{AR} \right]$$

$$\angle RAG = \angle RAB - \theta_2$$

$$\angle ARB = \pi - \angle ABC - \angle RAB$$

$$\angle ARQ = \frac{\pi}{2} - \angle ARB$$

$$\angle AQR = \pi - \arcsin \left[\frac{AR \sin \angle ARQ}{(R_1 + \tau)} \right]$$

$$\angle QAR = \pi - \angle AQR - \angle ARQ$$

$$\angle QAG = \angle QAR + \angle RAG$$

USE FOR TYPEWRITTEN MATERIAL ONLY

5.1.2

Slotted-Cone (Continued)

a. And $SAG \geq \theta_0$

Complete burnout has occurred.

Perimeter = 0.0

Area = 0.0

b. Or $SAG < \theta_0$

$$\angle ASB = \arcsin \left[\frac{(R_f - R_5 - \tau_w) \sin(\theta_2 + \angle SAG)}{(R_5 + \tau)} \right]$$

$$\angle SBA = \pi - \theta_2 - \angle SAG - \angle ASB$$

$$Area_1 = (\theta_0 - \angle SAG) R_f^2 + R_f (R_f - R_5 - \tau_w) \sin(\theta_2 + \angle SAG)$$

$$\angle RQA = \pi - \arcsin \left[\frac{AR \sin(\pi/2 - \angle ARB)}{(R_1 + \tau)} \right]$$

$$\angle RAQ = \frac{\pi}{2} + \angle ARB - \angle RQA$$

$$\angle QAG = \angle RAQ + \angle RAG$$

$$\angle RBS = \angle SBA - \angle ABC$$

$$RS = \angle RBS (R_5 + \tau)$$

(1) and If $\angle RAG \geq \theta_0$ go to 4a(2) or If $\angle RAG < \theta_0 \leq \angle QAG$ go to 4b(1)(3) or If $\angle RAG < \theta_0 > \angle QAG$ go to 4b(2)6. If $R_2 < \tau \geq \text{Temp}_{76}$, and If $T'_{V6} = T'_{V7}$

Complete burnout has occurred.

Perimeter = 0.0

Area = 0.0

USE FOR TYPEWRITTEN MATERIAL ONLY

5.1.2

Slotted-Cone (Continued)

7. If $R_2 < \tau \geq \text{TEMP}_{76}$, and if $T_{V6} > T_{V7}$, and $\tau \leq \tau_w$ (Figure 5.12),

$$\angle QAB = \arccos \left[\frac{(R_{V1} + \tau)^2 + (R_f - R_5 - \tau_w)^2 - (R_5 + \tau)^2}{2(R_1 + \tau)(R_f - R_5 - \tau_w)} \right]$$

$$\angle AQB = \arcsin \left[\frac{(R_f - R_5 - \tau_w) \sin(\angle QAB)}{(R_5 + \tau)} \right]$$

$$\angle ABQ = \pi - \angle AQB - \angle QAB$$

$$\angle ASB = \arcsin \left[\frac{(R_f - R_5 - \tau_w) \sin(\theta_2)}{(R_5 + \tau)} \right]$$

$$\angle ABS = \pi - \theta_2 - \angle ASB$$

$$\angle QAG = \angle QAB - \theta_2$$

- a. and $\angle QAG \geq \theta_0$

$$\angle AQB = \arcsin \left[\frac{(R_f - R_5 - \tau_w) \sin(\theta_2 + \theta_0)}{(R_5 + \tau)} \right]$$

$$\angle ABQ = \pi - \theta_2 - \theta_0 - \angle AQB$$

$$\angle QBS = \angle ABS - \angle ABQ$$

$$\text{PERIM} = \angle QBS (R_5 + \tau)$$

- b. or $\angle QAG < \theta_0$

$$\text{PERIM} = (\angle ABS - \angle ABQ) (R_5 + \tau) + (\theta_0 - \angle QAG) (R_1 + \tau)$$

USE FOR TYPEWRITTEN MATERIAL ONLY

5.1.2

Slotted-Cone (Continued)

8. If $R_2 < T \geq \text{Temp}_{76}$, and if $T'_{V6} > T'_{V7}$ and $\tau > \tau_w$ (Figure 5.12),

$$\angle SAB = \arccos \left[\frac{R_f^2 + (R_f - R_5 - \tau_w)^2 - (R_5 + \tau)^2}{2 R_f (R_f - R_5 - \tau_w)} \right]$$

$$\angle SAG = \angle SAB - \theta_2$$

a. and $\angle SAG \geq \theta_0$

Complete burnout has occurred.

Perimeter = 0.0
Area = 0.0

b. or $\angle SAG < \theta_0$

$$\angle QAB = \arccos \left[\frac{(R_1 + \tau)^2 + (R_f - R_5 - \tau_w)^2 - (R_5 + \tau)^2}{2 (R_1 + \tau) (R_f - R_5 - \tau_w)} \right]$$

$$\angle QAG = \angle QAB - \theta_2$$

$$\angle ABS = \arccos \left[\frac{(R_f - R_5 - \tau_w)^2 + (R_5 + \tau)^2 - R_f^2}{2 (R_f - R_5 - \tau) (R_5 + \tau)} \right]$$

(1) and if $\angle QAG \geq \theta_0$

$$\angle AQB = \arcsin \left[\frac{(R_f - R_5 - \tau_w) \sin(\theta_2 + \theta_0)}{(R_5 + \tau)} \right]$$

$$\angle ABQ = \pi - \theta_2 - \theta_0 - \angle AQB$$

$$\angle QBS = \angle ABS - \angle ABQ$$

$$QS = \angle QBS (R_5 + \tau)$$

$$\text{Perimeter} = QS$$

$$\begin{aligned} \text{Area} = & (\theta_2 + \theta_0 - \angle SAB) R_f^2 + R_f (R_f - R_5 - \tau_w) \sin \angle SAB \\ & - (R_5 + \tau)^2 \angle QBS - (R_f - R_5 - \tau_w) (R_5 + \tau) \sin \angle ABQ \end{aligned}$$

USE FOR TYPEWRITTEN MATERIAL ONLY

5.1.2.

Slotted-Cone (Continued)

(2) Or If $\angle QAG < \theta_0$

$$\angle ABQ = \arccos \left[\frac{(R_f - R_5 - \tau_w)^2 + (R_5 + \tau)^2 - (R_1 + \tau)^2}{2(R_f - R_5 - \tau_w)(R_5 + \tau)} \right]$$

$$\angle QBS = \angle ABS - \angle ABQ$$

$$QS = \angle QBS (R_5 + \tau)$$

$$\angle PAQ = \theta_2 + \theta_0 - \angle QAB$$

$$PQ = \angle PAQ (R_1 + \tau)$$

$$\text{Perimeter} = PQ + QS$$

$$\begin{aligned} \text{Area} = & (\theta_2 + \theta_0 - \angle SAB) R_f^2 + R_f (R_f - R_5 - \tau_w) \sin \angle SAB \\ & - (R_5 + \tau)^2 \angle QBS - (R_1 + \tau) (R_f - R_5 - \tau_w) \sin \angle QAB \\ & - (\theta_2 + \theta_0 - \angle QAB) (R_1 + \tau)^2 \end{aligned}$$

9. If $R_2 < \tau \geq \text{Temp}_{76}$, and if $T'_{V6} < T'_{V7}$ (Figure 5.13),

$$AT = \left[(R_f - R_5 - \tau_w)^2 + (R_5 + \tau)^2 - 2(R_f - R_5 - \tau_w)(R_5 + \tau) \cos \angle ABC \right]^{1/2}$$

$$\angle ATB = \arcsin \left[\frac{(R_f - R_5 - \tau_w) \sin \angle ABC}{AT} \right]$$

$$\angle STA = \pi/2 - \angle ATB$$

$$\angle AST = \pi - \arcsin \left[\frac{AT \sin \angle STA}{R_f} \right]$$

$$\angle TAB = \pi - \angle ABC - \angle ATB$$

$$\angle SAT = \pi - \angle AST - \angle STA$$

$$\angle SAG = \angle SAT + \angle TAB - \theta_2$$

a. And $\angle SAG \geq \theta_0$

Complete burnout has occurred.

5.1.2 Slotted-Cone (Continued)

$$\text{Perimeter} = 0.0$$

$$\text{Area} = 0.0$$

$$\text{b. Or } \angle \text{SAG} < \theta_0$$

$$\angle \text{QSA} = \pi - \angle \text{AST}$$

$$\angle \text{AQS} = \pi - \arcsin \left[\frac{R_f \sin \angle \text{QSA}}{(R_1 + \tau)} \right]$$

$$\angle \text{QAS} = \pi - \angle \text{AQS} - \angle \text{QSA}$$

$$\angle \text{QAG} = \angle \text{SAG} + \angle \text{QAS}$$

$$(1) \text{ And if } \angle \text{QAG} \geq \theta_0$$

$$\angle \text{QAS} = \theta_0 - \angle \text{SAG}$$

$$\angle \text{AQS} = \pi - \angle \text{QAS} - \angle \text{QSA}$$

$$\text{QS} = \frac{R_f \sin \angle \text{QAS}}{\sin \angle \text{AQS}}$$

$$\text{Perimeter} = \text{QS}$$

$$\text{Area} = \angle \text{QAS} R_f^2 - R_f \text{QS} \sin \angle \text{QSA}$$

$$(2) \text{ Or if } \angle \text{QAG} < \theta_0$$

$$\angle \text{PAQ} = \theta_0 - \angle \text{QAG}$$

$$\text{PQ} = (R_1 + \tau) \angle \text{PAQ}$$

$$\text{QS} = R_f \frac{\sin \angle \text{QAS}}{\sin \angle \text{AQS}}$$

$$\text{Perimeter} = \text{PQ} + \text{QS}$$

$$\text{Area} = (\angle \text{PAQ} + \angle \text{QAS}) R_f^2 - \text{QS} R_f \sin \angle \text{QSA}$$

$$- \angle \text{PAQ} (R_1 + \tau)^2$$

USE FOR TYPEWRITTEN MATERIAL ONLY

5.2

Grain Longitudinal Geometry

Representative longitudinal configurations which can be accommodated by the computer program have been shown in Figure 2.2. Many configurations are possible: monolithic or segmented grains in the cylindrical section of the motor; head-end with web or a straight through grain in the fore-head section; straight through grain in the aft-head; external and internal taper; and elliptical contour fore-head and aft-head motor sections are examples. Grain cross sectional geometry can be varied within the cylindrical section of the motor.

5.2.1

Head-End with Web, Fore-Head Section

This section describes the analysis of the head-end with web which is solved in the third control routine of the program. Description of the head-end with web calculations are based on the motor fore-head section shown in Figure 5.14. Grain geometry within the fore-head section is based on the input geometrical cross section at the forward tangent plane.

The burning surface area for all distances burned and the initial propellant volume are obtained by separating the analysis into four blocks. Each block performs the following function:

Block 1 calculates the surface area versus distance burned and the initial volume of the propellant tip. The volume and area are obtained by integrating elemental areas and volumes consisting of trapezoids and parallelograms.

Block 2A calculates the surface area of the pseudoellipsoid minus the igniter opening using the Theorem of Pappus. Block 2B calculates the volume of propellant between the inner and outer ellipsoids and the surface area on the pseudo ellipsoid which is covered by the propellant tip. The volume and area are obtained by integrating elemental areas and volumes using the Theorem of Pappus.

Block 3 calculates the initial total propellant volume. The volume between the inner and outer ellipsoids is determined from the difference in volume of two oblate spheroids. The volume of the tip is obtained by adding the volumes of cylindrical segment elements.

Total surface area, A , versus distance burned is obtained by combining the blocks:

$$A = A_{\text{Block 1}} + A_{\text{Block 2A}} - A_{\text{Block 2B}}$$

The initial propellant volume is calculated in two ways, one

5.2.1

Head-End with Web, Fore-Head Section (Continued)

method as in Block 3 and the other by:

$$\text{Volume} = V_{\text{Block 1}} + V_{\text{Block 2B}}$$

The initial volumes are compared in subroutine VOLSUB and a correction factor, A_R , is obtained which is added to the total burning surface area. This is due to the assumption that the difference is caused by the integration method in the Block 1 analysis.

Subroutine HDNSUB

Subroutine HDNSUB is the control routine which sets up the correct variables and equations to perform the block 1, 2A, 2B, and 3 analysis.

5.2.1.1

Block 1 Analysis

The generalized forked wagon wheel grain configuration is divided into 13 sectors (1, 2, 3, 3A, 3B, 4, 5, 6, 7, 9, 10, 11, 11A, 11B, 12, 13) as shown in Figure 5.15. Sectors (1, 2, 4, 5, 6, 7, 9, 10, 12 and 13) share a common analysis, as do sectors (3 and 11) and (3A, 3B, 11A, and 11B); however, some of the sectors have special equations for line segments and angles. The sector boundaries used for the head-end with web analysis, Figure 5.15, are different than sector boundaries used for the straight through grain analysis, Section 5.2.3 and Figure 5.32.

Subroutine SCI is the control routine to determine the surface area versus distance burned and initial volume of the propellant tip for the block 1 analysis and is called from subroutine HDNSUB with an argument L to indicate the sector to be computed. The analysis for sectors 1 and 3 will be explained in detail since the analysis for the other sectors is similar.

Figure 5.16 shows geometric constructions and calculation control planes used to obtain elemental burn surface areas and propellant volumes in the propellant tip. Also shown is part of a forked wagon wheel grain and the required constructions. For clarity, only one tip is shown. Two planes, "A" and "B", control volume and area calculations for the propellant tip. These two planes are generated as follows: The intersection of the surface of the propellant tip with the forward tangent plane, i.e., the tangent plane perimeter, Figure 5.16b, is divided into incremental lengths, ΔL . Lines, spaced ΔL apart, perpendicular to the tangent plane, are run from the tangent plane along the side of the propellant tip to intersect the inner ellipsoid. These intersections locate points P_{oa} and P_{ob} .

USE FOR TYPEWRITTEN MATERIAL ONLY

5.2.1.1 Block 1 Analysis (Continued)

Perpendiculars to the side of the propellant tip from points P_{oa} and P_{ob} locate points P_{1a} and P_{1b} on the plane of symmetry or on the outer ellipsoid depending on the initial location of each ΔL_0 along the propellant tip perimeter in the tangent plane. Figure 5.16a shows P_{1a} and P_{1b} on the plane of symmetry. Perpendiculars to the inner ellipsoid at points P_{oa} and P_{ob} locate points P_{2a} and P_{2b} on the outer ellipsoid. Points P_{oa} , P_{1a} , P_{2a} , P_{ob} , P_{1b} , and P_{2b} define the A and B planes. Points P_{3a} and P_{3b} lie in the A and B planes and are located on the outer ellipsoid.

The trapezoid formed by ΔL_0 on the perimeter of the tip at the tangent plane and points P_{oa} and P_{ob} is the area calculation element.

Perpendiculars, in "x-z" planes, for these same four points form the volume calculation elements which are bounded by the trapezoidal area elements as shown in Figure 5.16. The ΔL_0 spacing on the propellant tip perimeter which governs plane placement varies in value along the perimeter as follows:

The ΔL_0 spacing, Figure 5.16b, is calculated first as:

a) ΔL_0 of each sector is:

$$\Delta L_0 = \left(\frac{\Delta L}{R_f} \right) R_f$$

where $\frac{\Delta L}{R_f}$ is an input parameter.

b) ΔL_0 is then modified by:

$$\Delta L_0 = \frac{K(\Delta L_0 \text{ previous})}{(D_{pr} + D_{ps})}$$

where

D_{pr} and D_{ps} are the distances between planes A and B, Figure 5.17.

The starting value of ΔL_0 previous is ΔL_0 .

5.2.1.1 Block 1 Analysis (Continued)

The factor K , an input variable, is the major parameter in determining the distance between the two planes. The incremental distance ΔL_0 is stepped along the sector perimeters from $L_x = 0$ to $L_x = L_p$.

5.2.1.1.1 Sector 1 (Figure 5.15)

Subroutines ASUBC, BSUBC, RASUBB, XRSUBB, THETAR, GAMSUB, GAMA2S, POSUB, PLSUB, P3SUB, ROPSB, and VSTRSB are used to obtain the surface area and initial volume for sector 1.

The distance D_{03} between the points P_0 and P_3 on the inner ellipsoid are shown in Figure 5.18 for planes that are located at increments of one-tenth the perimeter length of a sector. A minimum perimeter length, HOLDR, is set equal to the perimeter length in which the distance D_{03} for the plane is greater than the burn distance, τ . The first plane is located at the perimeter length HOLDR and subsequent planes are spaced the distance ΔL_0 apart. HOLDR is initialized to the perimeter length for plane B on subsequent iterations.

After the minimum perimeter length, HOLDR, is determined, the distance D_{03} for a plane located at the total perimeter $AL(1)$ for the sector is determined. The parameter TDMAX is initialized to the maximum value of TDMAX or D_{03} for the sector and then checked with the maximum permissible burn distance, TAUM. If TDMAX equals or exceeds TAUM, an error condition exists and execution is terminated. If all D_{03} 's are less than TAUM, the radial burning surface area between the planes and the burning surface area on the propellant tip are determined.

Radial burning surface area, Figure 5.17:

$$ASI = \frac{(L_{TA} + L_{TB})(D_{pr} + D_{ps})}{4}$$

Propellant tip burning surface area:

$$ASI = \frac{(L_p - HOLDR)(R_2 - \tau)(Y_{0A} + Y_{0B})}{2R_2}$$

where Y_0 is defined in Figure 5.21.

USE FOR TYPEWRITT. MATERIAL ONLY

5.2.1.1.1 Sector 1 (Figure 5.15) (Continued)

L_p is the perimeter length along the initial grain perimeter measured from the beginning of a sector to a general point. $HOLDR$ is the prior incremental value of L_p .

The initial volume is determined from the following equation:

$$V_{STR} = \frac{(Y_{OA} + Y_{OB})(L_p - HOLDR)R_2}{4}$$

Subroutine ASUBC

Subroutine ASUBC sets up the correct variables and equations to determine the coordinates (X, Y, and Z) of the points P_{oa} , P_{la} , and P_{3a} for planes located at increments of one-tenth L_p along the perimeter of the sector. Subroutines RASUBB, XRSUBB, THETAR, GAMSUB, GAMA2S, POSUB, PISUB, P3SUB, and TRAN are called to determine the coordinates.

Subroutine RASUBB

Subroutine RASUBB determines the length of the radius vector, R_{aT} , from the motor axis to a general point in a sector. The perimeter length from the beginning of the sector to a general point is required in calculating a value of R_{aT} for each sector, Figure 5.19.

Subroutine XRSUBB

Subroutine XRSUBB determines the X-coordinate, X_{ra} , for a point located on the perimeter of a sector. The parameter R_{aT} from the RASUBB subroutine is required to obtain the coordinate. A separate equation is required for each sector, Figure 5.15.

Subroutine THETAR

Subroutine THETAR determines the angle, θ_r , between the Z-axis and the line segment R_{aT} . The parameters X_{ra} and R_{aT} are required to obtain θ_r , Figure 5.15.

Subroutine GAMSUB

Subroutine GAMSUB determines the angle γ_1 between the line normal to the perimeter in the XZ-plane, and a line normal to the line segment R_{aT} , Figure 5.19. The perimeter length of the sector and the angle θ_r are required to obtain γ_1 . A separate equation for each of the 16 individual sectors is required. γ_1 for sector 3A is not equal to γ_1 for sector 3B (also true for sectors 11A and 11B).

5.2.1.1.1 Sector 1 (Figure 5.15) (Continued)

Subroutine GAMA2S

Subroutine GAMA2S determines the angle γ_2 between the Y-axis and a line normal to the ellipse $\left(\frac{Y}{B_{oe}}\right)^2 + \left(\frac{Z'}{A_{oe}}\right)^2 = 1$, which is defined by the ellipse ratio β_{oe} , at the point $Z' = RAT$. Although γ_2 is in the rotated YZ' -plane and not in the YZ -plane, because of the symmetry of the ellipsoid about the Y-axis, computation can occur in the YZ -plane, Figure 5.20.

Subroutine POSUB

Subroutine POSUB determines the coordinates X, Y, and Z of the point P_0 which is located at the intersection of the inner ellipse and a line extended along the propellant surface parallel to the tip from a general point on the perimeter, Figures 5.16 and 5.21.

The parameters R_{aT} and θ_r are required to determine the coordinates. The coordinates X_0 and Z_0 are obtained by trigonometry from R_{aT} and θ_r , Figure 5.21. The equation of the ellipsoid:

$$\left(\frac{Y_0}{B_{oe}}\right)^2 + \left(\frac{X_0^2 + Z_0^2}{A_{oe}^2}\right) = 1$$

yields the coordinate Y_0 .

The remaining points defining the plane (P_1 , P_2 , and P_3) are obtained from P_0 . Radial burning in the plane originates at P_0 .

Subroutine P1SUB

Subroutine P1SUB determines the coordinates (X_1 , Y_1 , and Z_1) of the point P_1 which is located on the Y-Z plane or on the outer ellipse along a line through point P_0 and normal to the sector perimeter as shown in Figure 5.22. P_1 is normally located on the Y-Z plane; however, for sectors 7 and 9, when the angle γ_1 exceeds β_{7lmax} in sector 7 or β_{9lmax} in sector 9, the point P_1 is located on the outer ellipse and is coincident with P_3 . The line segment Z_{1aT} shown in Figure 5.19 is used to flag subroutine P3SUB that P_1 is coincident with P_3 whenever Z_{1aT} is non-zero.

5.2.1.1.1 Sector 1 (Figure 5.15) (Continued)

β_{9lmax} and β_{7lmax} are geometry plane constants discussed in Section 5.1.1.

Subroutine P3SUB

Subroutine P3SUB determines the coordinates (X_3 , Y_3 , and Z_3) of the point P_3 that is located on the outer ellipse. If Z_{1aT} , determined in subroutine P1SUB, is not equal to zero, P_3 is coincident with P_1 and the coordinates of P_3 are set equal to the coordinates of P_1 . If Z_{1aT} is equal to zero, then P_3 is in the Y-Z plane and the coordinates of P_3 are determined from the angle Φ shown in Figure 5.22.

Subroutine ROPSB

Subroutine ROPSB sums the values of the Y axis coordinate of P_o (Y_{OA} and Y_{OB}) or P_o' (Y_{OA}' and Y_{OA} or Y_{OB}' and Y_{OB}) in planes A and B which are used in subroutine SCI to compute the surface area along the grain face. If the burn distance τ is greater than the maximum permissible burn distance for the sector, then the value of the sum is set to zero.

Subroutine BSUBC

Subroutine BSUBC sets up the correct variables and equations to determine the coordinates (X, Y, and Z) of the points P_{ob} , P_{1b} , and P_{3b} located at increments of ΔL_0 along the perimeter of sector 1, Figure 5.15. The A plane coordinates are initialized to the prior B plane coordinates during the integration along the perimeter.

Subroutine VSTRSB

Subroutine VSTRSB determines the initial volume of the sectors. The incremental volume for each sector is determined from the incremental cross-sectional areas of the sectors and the average of the P_o point Y-coordinates of the A and B planes.

5.2.1.1.2 Sector 3

Subroutines HASUBC, HBSUBC, HAPSB, HBPSBC, DPRASB, RASUBB, XRSUBB, THETAR, GAMSUB, GAMA2S, POSUB, P1SUB, P3SUB, and ROPSB are used to obtain the surface area and initial

USE FOR TYPEWRITTEN MATERIAL ONLY

5.2.1.1.2 Sector 3 (Continued)

volume for sectors 3, 3A, and 3B. As in sector 1, two planes are used to obtain the surface area. The distance between the planes A and B is determined by ΔL_0 in the same manner as for sector 1. The analysis for the surface area in sector 3 is similar to the analysis for the surface area in sector 1 except that the A and B planes will cross as shown in Figure 5.23 for sectors 3 and 11 and in Figure 5.24 for sectors 3A, 3B, 11A, and 11B.

The perimeter length of sector 3A (AL_{3A}) is determined from the geometry constants, and the coordinates of points P_0 , P_1 , and P_3 for both planes A and B. Since the surface area of sector 3B is identical with the surface area of sector 3A, the grain surface areas of both sectors between the planes are determined simultaneously by stepping the B plane an increment ΔL_0 along the sector 3A perimeter. The surface area along the grain between the planes is determined from the following equation:

$$ASI = \frac{(Y_0 + Y_0')}{2} \left[(X_0 - X_0')^2 + (Z_0 - Z_0')^2 \right]^{1/2}$$

The surface area along the tip for sector 3 is determined from the following equation after the integration for sector 3A is complete:

$$ASI = L_3 \frac{(R_2 - \tau)}{R_2} \frac{(Y_{0A} + Y_{0B})}{2}$$

where L_3 is the initial sector 3 perimeter length.

The initial volume of sector 3, and sectors 3A and 3B, is determined at the beginning of subroutine SCTOR1 after completion of the block 1 analysis as follows:

$$V_{STR} = \frac{(Y_{0A} + Y_{3A} + Y_{0B} + Y_{3B})(\tau_{2max} + R_3)}{4} + \frac{(\alpha_{01} - \alpha_{02})(Y_{0B} + Y_{3B})R_3^2}{4}$$

Subroutines RASUBB, XRSUBB, THETAR, GAMSUB, GAMA2S, POSUB, PLSUB, P3SUB, and ROPSB have been explained in Section 5.2.1.1.1 for sector 1. Therefore, only the subroutines unique to sector 3 are explained in this section.

5.2.1.1.2 Sector 3 (Continued)

Subroutines HASUBC, HBSUBC, HAPSUBC, and HBPSUBC

The function of these subroutines is the same as the function of subroutines ASUBC and BSUBC for sector 1; the correct variable and equations are set up to determine the coordinates (X, Y, and Z) of the points P_0 , P_1 , and P_3 for planes located at increments of $\Delta L0$ along the perimeter of the sector. Subroutine HASUBC deals with plane A, subroutine HBSUBC deals with plane B, subroutine HAPSBC deals with plane A', and subroutine HBPSBC deals with plane B', Figure 5.24.

Subroutine DPRASB determines the distance between the points P_{ra} and P'_{ra} and between P_{sa} and P'_{sa} that lie on the planes produced in sector 3A as shown in Figure 5.24. These distances DP_{ra} and DP_{sa} are used in subroutine SCI to alter the distance between the planes ($\Delta L0$) for each successive iteration as explained in Section 5.2.1.1 for sector 1.

5.2.1.2 Block 2A Analysis

Subroutine SCTOR1 is the control routine to determine the surface area of the pseudoellipsoid for the block 2A analysis and is called from subroutine HDNSUB. The initial volume of propellant for sectors 3A, 3B, 11A, and 11B is determined prior to the integration for the pseudoellipsoid surface area.

Points P_0 and P_1 that are used in Block 2A are not the same plane points used in the Block 1 analysis. Points P_0 and P_1 lie on the inner ellipsoid and are spaced a distance DS along the ellipsoid, Figure 5.25, starting at the igniter opening. The spacing of P_0 and P_1 are determined as follows:

5.2.1.2 Block 2A Analysis (Continued)

First

$$\Delta R = \left(\frac{\Delta R_v}{R_f} \right) R_f$$

where

$$\frac{\Delta R_v}{R_f} \text{ is an input parameter}$$

 R_f grain radius at the forward tangent plane.

Then each successive increment is

$$\Delta R = \frac{KK}{DS} \Delta R$$

where KK is an input.

Initially, if the motor has an igniter hole in the head end, Z_{p0} , which is the Z-coordinate of the point, P_0 , is set equal to R_{ig} . Otherwise, Z_{p0} is set equal to zero. ΔR is then added to Z_{p0} to obtain Z_{p1} , which is the Z coordinate of the point P_1 . Subroutine YPSUB is then called to obtain the Y coordinate of points P_0 and P_1 . The radius of curvature ρ_1 at the point P_1 on the pseudoellipsoid, the length, b , of the Y-intercept of the normal line to the ellipse at P_0 , and the Z coordinate Z_1 are then determined to obtain the arc length, L_0 . Using these values, the incremental strip surface area is determined. This procedure is then repeated by setting $Z_{p0} = Z_{p1}$, and $Z_{p1} = Z_{p0} + \Delta R$ until $Z_{p1} = R_f - \tau_w$. The area adjacent to the igniter hole is then determined and added to the incremental sum.

Subroutines YPSUB, ROELSB, LBSUB, ZISUB, and AIGSUB are used to determine the surface area of the pseudoellipsoid. The surface area for an incremental strip, as shown in Figure 5.25, is determined by the following equation:

$$AS! = \frac{\pi L_0}{2N0} (Z_{p1} + \tau \sin \alpha_{rc1} + Z_{p0} + \tau \sin \alpha_{rc0})$$

5.2.1.2

Block 2A Analysis (Continued)

The surface area for the entire pseudoellipsoid, including the igniter hole, is obtained by summing the incremental areas. The surface area around the igniter hole is added to the sum of the incremental areas. The lateral surface area of the igniter hole is assumed to be a non-burning inhibited surface.

Subroutine YPSUB

Subroutine YPSUB determines the Y coordinate of a point on the pseudoellipsoid. The angles α_{rc0} and α_{rc1} of the points P_{0a} and P_{0b} as shown in Figure 5.25 are also determined.

Subroutine ROE1SB

Subroutine ROE1SB determines the radius of curvature ρ_1 at the point P_0 on the inner ellipsoid. The radius of curvature is determined by the standard equation:

$$\rho_1 = \frac{[1 + (z')^2]^{3/2}}{z''}, \quad \text{where}$$

$$z' = \frac{dz}{dy}, \quad z'' = \frac{d^2z}{dy^2}$$

Subroutine LBSUB

Subroutine LBSUB determines the length b of the Y-intercept of the line normal to the inner ellipse at P_0 , Figure 5.25. The length is used in subroutine ZISUB to obtain the coefficients of the ellipse equation which defines the pseudoellipsoid.

Subroutine ZISUB

Subroutine ZISUB determines the Z-coordinate produced by the intersection of the outer ellipse:

$$\left(\frac{Y}{B_{1e}}\right)^2 + \left(\frac{Z}{A_{1e}}\right)^2 = 1$$

and the normal line to the ellipse at P_1 .

5.2.1.2 Block 2A Analysis (Continued)

Subroutine AIGSUB

Subroutine AIGSUB determines the surface area around the igniter opening. The area is obtained by finding the surface area of revolution which requires an angle π/N_0 , an arm length

$$\frac{\tau \sin \alpha_{rN}}{2} + R_{ig}, \text{ and an arc length } \tau \alpha_{rN}, \text{ Figure 5.26.}$$

Initially, the Y-intercept of a line normal to the inner ellipse at the igniter radius is obtained,

$$Y_{NO} = \left[B_{OE}^2 - \frac{R_{ig}^2}{\beta_{OE}^2} \right]^{1/2}$$

from which the angle between the Y-axis and the igniter opening on the inner ellipse is obtained,

$$\alpha_{rN} = \frac{\pi}{2} - \arccos \left[\frac{R_{ig}}{\left[(\beta_{OE}^2 Y_{NO})^2 + R_{ig}^2 \right]^{1/2}} \right]$$

and then the surface area of revolution is determined (Theorem of Pappus)

$$A_{ig} = (2R_{ig} + \tau \sin \alpha_{rN}) \frac{\pi \tau \alpha_{rN}}{2 N_0}$$

5.2.1.3 Block 2B Analysis

Subroutine SCTOR2 is the control routine to determine the surface area on the pseudoellipsoid that is covered by the propellant tips for the block 2B analysis. The surface area is determined by dividing the cross-sectional grain configuration into 12 sectors in subroutine HDNSUB, as shown in Figure 5.27. The surface area on the pseudoellipsoid for each sector is then determined by subroutine S2SK. Summing the surface areas for all sectors, and subtracting this area from the area obtained in blocks 1 and 2A, yields the total surface area of the head-end with web for any value of τ .

5.2.1.3 Block 2B Analysis

Subroutine S2SK

Subroutine S2SK determines the sector surface area on the pseudoellipsoid and is called from subroutine SCTOR2 with an argument to indicate the current sector. Each sector is set up to determine the incremental surface area contributed by that sector. Thus, in sector 3, Z_{p0} is set equal to R_{p2} , then R_{p3} is obtained and θ_{r0} is set equal to θ_{r1} . ΔR is then added to Z_{p0} to obtain Z_{p1} as shown in Figure 5.28. Y_{p0} , Y_{p1} and DS are then determined. The spacing for ΔR is calculated in the same manner as was done in Block 2A.

Finally, the parameters L_{pp} , X_{r1} , θ_{r0} , θ_{r1} , α_{rc0} , α_{rc1} , ρ_1 , b , Z_1 , and L_Q are determined and the incremental surface area for the sector is obtained from the equation, Figure 5.25:

$$L_Q = (\rho_1 + \tau)(\alpha_{rc1} - \alpha_{rc0})$$

and

$$AS = L_Q(Z_{p1} + \tau \sin \alpha_{rc1} + Z_{p0} + \tau \sin \alpha_{rc0}) \frac{(\theta_{r1} + \theta_{r0})}{4}$$

Z_{p0} is then set to Z_{p1} and the procedure repeated until Z_{p1} equals maximum radius of the sector.

The initial volume between the inner and outer ellipsoids for each sector is determined as follows (see Figure 5.25):

$$V_{ST0} = \left[(\rho_1 + D_{WB})^2 - \rho_1^2 \right] (\theta_{r1} + \theta_{r0})(\alpha_{rc1} - \alpha_{rc0}) \frac{Z_G N_0}{2}$$

where:

$$D_{WB} = \frac{Z_1 - Z_{p1}}{\sin \alpha_{rc1}}$$

$$D_{S1} = \rho_1(\alpha_{rc1} - \alpha_{rc0})$$

$$D_{S2} = (\rho_1 + D_{WB})(\alpha_{rc1} - \alpha_{rc0})$$

5.2.1.3 Block 2B Analysis (Continued)

$$X_{WB} = \frac{(2D_{S2} + D_{S1}) D_{WB}}{3(D_{S2} + D_{S1})}$$

$$Z_G = \frac{(Z_{P0} + Z_{P1})}{2} + X_{WB} \sin \left[\frac{\alpha_{rc1} + \alpha_{rc0}}{2} \right]$$

Subroutines YPSUB, XRTHR, ROE1SB, LBSUB, and ZISUB are used to obtain the surface area. All of these subroutines except XRTHR have been explained in Section 5.2.1.2 for the block 2A analysis.

Subroutine XRTHR

Subroutine XRTHR is a set-up subroutine that uses subroutine XRSUBB to obtain the X-coordinate of a point located on the perimeter of a sector shown in Figure 5.28. The angle θ_{r1} between the Z axis and a line from the motor axis to the point on the sector is also determined.

5.2.1.4 Block 3 Analysis

Subroutine VOLSUB is the control routine which determines the initial propellant volume for the block 3 analysis. Initially, the volume present in the web region is determined from the volume produced by the difference of volumes of two oblate spheroids minus the volume of the igniter hole as follows:

$$V_{EH} = \frac{2}{3} \pi (B_{1e} A_{1e}^2 - B_{oe} A_{oe}^2) - \pi R_{ig}^2 (B_{1e} - B_{oe})$$

Next, the volume of the propellant tips is determined by dividing the grain configuration into 12 sectors as was done in block 2B, Figure 5.27. The volume contributed by each sector (subroutine VSEC) is then added to V_{EH} to obtain the total volume of propellant in the head-end, V_{FH} . Finally, the surface area generated by radial burning from the line of intersection of the propellant tip with the head-end web, A_R , is approximated in the following manner:

The propellant volume in the radial burning portion is determined,

$$V_R = V_{FH} - V_{EH} + V_{STO} - 2 V_{STR} N_0$$

5.2.1.4

Block 3 Analysis (Continued)

The value for thickness burned at which the maximum surface area of this radial burning portion occurs is assumed to be the thickness at which the initial burnout of the web portion of the head-end web begins (for most cases this is very nearly correct). The curve for surface area versus thickness burned for this fuel volume is assumed to have the following equations (Figure 5.29):

$$A_R = \begin{cases} \frac{(T_{\text{DMAX}} - \tau_{\text{WH}})^q \tau}{\tau_{\text{WH}}} & \text{for } \tau \leq \tau_{\text{WH}} \\ (T_{\text{DMAX}} - \tau)^q & \text{for } \tau_{\text{WH}} < \tau < T_{\text{DMAX}} \\ 0 & \text{for } \tau \geq T_{\text{DMAX}} \end{cases}$$

where

T_{DMAX} = maximum burn distance in the head-end section

τ_{WH} = maximum web thickness in the head-end section

q = calculated exponent

The maximum thickness burned in the head-end web is determined,

If $B_{\text{IE}} - B_{\text{OE}} > \tau_w$, $\tau_{\text{WH}} = \tau_w$

If $B_{\text{IE}} - B_{\text{OE}} \leq \tau_w$, $\tau_{\text{WH}} = B_{\text{IE}} - B_{\text{OE}}$

The radial burning volume is matched to the following integral ($V_R = V_{\text{RX}}$) by an iteration process to determine the exponent q ,

$$V_{\text{RX}} = \int_0^{\tau_{\text{WH}}} \frac{(T_{\text{DMAX}} - \tau_{\text{WH}})^q}{\tau_{\text{WH}}} \tau d\tau + \int_{\tau_{\text{WH}}}^{T_{\text{DMAX}}} (T_{\text{DMAX}} - \tau)^q d\tau$$

or

5.2.1.4 Block 3 Analysis (Continued)

$$V_{RX} = \frac{(\tau_{JMAX} - \tau_{WH})^q \tau_{WH}}{2} + \frac{(\tau_{JMAX} - \tau_{WH})^{q+1}}{q+1}$$

The above integral represents the area beneath the curve shown in Figure

The radial burning area A_R is added to the total surface area, ASI, at the completion of the block 3 analysis in subroutine VOLSUB.

Subroutine VSEC

Subroutine VSEC determines the initial volume of the tips that are present in the fore-head and is called from subroutine VOLSUB with an argument to indicate the sector to be computed. The same subroutines used in the block 2B analysis to obtain the incremental surface area of a pseudoellipsoid are used in this subroutine to obtain the incremental sector volumes. An arc length L_Q is obtained from the expression

$$L_Q = Z_{P1} - Z_{P0}$$

and the incremental volume within a sector is obtained from the expression

$$\Delta V = NO L_Q (Z_{P1} + Z_{P0}) (\theta_{ri} + \theta_{r0}) (Y_{P0} + Y_{P1}) / 4$$

5.2.2 Cylindrical Section

The longitudinal cylindrical section is that portion of the motor between the forward and aft tangent planes, Figure 4.3. Its length is designated as h_{CO} and the radius as R_f as shown in Figure 5.30. It may be either straight or tapered and may contain either a monolithic or segmented grain with or without a tapered port.

The longitudinal cylindrical section is first divided into incremental mass addition regions by locating increment dividing planes every delta Z distance, starting at the forward tangent plane and proceeding to the aft tangent plane. A maximum of 100 increment dividing planes is allowed.

The longitudinal cylindrical section is then divided by a number of reference planes. A minimum of two is required, one at each of the tangent planes. A maximum of eighteen is allowed to indicate changes in the cross-sectional grain geometry at specified locations within the section as shown in Figure 4.3. The cross sectional grain geometry is described at each reference plane used, by the port perimeter, L_p , the case radius, R_f , the port area, A_p , the propellant area, A_{FF} , the radius of gyration, K_{GY} , and the distance to the inert sliver, T_{SLVR} , if the inert

USE FOR TYPEWRITTEN MATERIAL ONLY

5.2.2 Cylindrical Section (Continued)

sliver option is used.

In segmenting propellant grains, the location of the slots, which separates the grain segments, is determined by defining the distance of the slot forward and aft interfaces from the forward tangent plane of the motor. This is shown by the values SA(1), SB(1), SA(2), SB(2), etc., Figure 4.3. The increment dividing planes, in this case, are determined as above, but with each segment treated as having a forward and aft tangent plane. The reference planes may be located within a slot, within a grain segment, or on a slot interface. The restrictions on the number of increment dividing planes and reference planes is also 100 and 18 respectively.

The following describes the subroutines which determine the longitudinal geometry for the cylindrical section:

Subroutine MNCHN4

Subroutine MNCHN4 contains the program control logic required to obtain the internal ballistic solution and the control logic used to initialize the working reference planes for subroutine SEGSUB. During burning, the distance burned at each reference plane for each time increment is determined by linear interpolation between adjacent increment dividing planes. When the reference plane is located within a slot, the increment dividing planes located within the grain segment are used for extrapolation to obtain the reference plane distance burned. When the distance burned has been determined for each input reference plane, the perimeter length, port area, and radius of gyration for each reference plane, and the burn area and CG location for each head section is determined from a table look-up procedure in the geometry tables.

During the internal ballistic solution for each time point the cylindrical section working reference planes (X and Y) are set up for successive input reference planes (1-2, 2-3, etc.).

Subroutine SEGSUB

Subroutine SEGSUB contains the program control logic which determines the perimeter length, cross-sectional propellant and port area, propellant volume, and mass generation at each increment dividing plane or mass addition region. The control logic is set to check for the existence of a slot forward interface between adjacent increment dividing planes, to check the location of the current upstream (X) and downstream (Y) working reference planes, and to set the parameters required in performing the gas dynamic solution for the slots and mass addition regions. Each increment dividing plane case radius, perimeter length, sliver radius, fuel area and port area and radius of gyration are obtained by linear interpolation between each reference plane.

5.2.3

Straight Through Grain, Motor End Sections

The following is applicable to motor aft-head or fore-head sections with straight through grains. A straight through grain is shown in Figure 5.31. The E subscript shown is set to N for analysis of the aft-head and to H for the fore-head.

Basic geometry constants required for analysis are calculated prior to determination of burn surface area versus distance burned and propellant volume. These constants are calculated in subroutine ENDCSB and are shown in Figure 5.30.

5.2.3.1

Geometry Constants

Subroutine ENDCSB

Initially, the case opening radius is determined from the input parameter DE1:

$$R_{E1} = \frac{D_{E1}}{2}$$

Then the angle between the tangent to the ellipse section at the radius R_{E1} and the motor axis is determined:

$$\alpha_{ER} = \arccos R_{E1} / [(R_f^2 - R_{E1}^2) \beta_E^2 + R_{E1}^2]^{1/2}$$

where R_f and β_E are input parameters.

If α_{ER} is less than or equal to the maximum allowable angle, $\alpha_{0E\max}$, defined by input, then $\alpha_{0E\max}$ is set to α_{ER} , R_{E2} is set to R_{E1} and h_{E1} is set to zero. If, however, α_{ER} is greater than $\alpha_{0E\max}$, then

$$R_{E2} = \frac{\cos(\alpha_{0E\max}) R_f \beta_E}{[\cos(\alpha_{0E\max})^2 \beta_E^2 - \cos(\alpha_{0E\max})^2 + 1.0]^{1/2}}$$

and

$$h_{E1} = (R_{E2} - R_{E1}) \frac{\sin(\alpha_{0E\max})}{\cos(\alpha_{0E\max})}$$

Next, the length of the head elliptical section is determined:

$$h_{E2} = [R_f^2 - R_{E2}^2]^{1/2} / \beta_E$$

5.2.3.1

Geometry Constants, Subroutine ENDGSB (Continued)

and the length of the end section is computed:

$$h_{EO} = h_{E1} + h_{E2}$$

If h_{EO} is less than the maximum burn distance at the adjacent tangent plane, τ_{max} , determined in Section 5.1, Geometry Constants, the end section is lengthened to the maximum burned distance:

$$h_{ER} = \tau_{max}$$

If, however, h_{EO} is greater than or equal to τ_{max} , then $h_{ER} = h_{EO}$. Finally, the maximum burn distance in the conical section and the end section is determined:

$$\tau_{E1} = [(R_{E2} - R_{E1})^2 + h_{E1}^2]^{1/2}$$

$$\tau_{EO} = [(R_f^2 - R_{E1})^2 + h_{EO}^2]^{1/2}$$

The complete end section case volume is determined from:

$$V_{CE} = (h_{ER} - h_{EO}) \pi R_f^2 + \frac{1}{3} [3 \frac{R_f^2}{\beta_E} - h_{E2}^2] [\pi h_{E2} \beta_E^2] \\ + (R_{E2} R_{E1} + R_{E1}^2 + R_{E2}^2) \pi \frac{h_{E1}}{3}$$

The following coefficients are calculated for use in subroutine RCSUB when the burning distance $\tau > \tau_{E1}$. They are the coefficients of a fourth degree equation of the intersection of an ellipse and a circle and are used to compute the location of intersection of the burning surface and the case wall.

$$CAE = (\beta_E^2 - 1)^2$$

$$CBE = -(\beta_E^2 - 1)^2 \beta_E^2 R_{E1}^4$$

$$CCCE = [\beta_E^4 R_{E1}^2 3 + (\beta_E^4 + \beta_E^2) h_{EO}^2 + R_f^2 (\beta_E^2 - 1) - R_E^2 \beta_E^2] 2$$

$$CCVE = 2(\beta_E^4 - \beta_E^2)$$

$$CDCE = [(h_{EO}^2 + R_{E1}^2) \beta_E^2 + R_f^2 \beta_E^2] 4R_{E1}$$

5.2.3.1 Geometry Constants, Subroutine ENDCSB (Continued)

$$CDVE = 4 R_{E1} \beta_E^4$$

$$CECE = R_f^4 + (R_{E1}^2 - h_{EO}^2) \beta_E^2 R_f^2 + (R_{E1}^2 + h_{EO}^2) \beta_E^4$$

$$CEVE = (h_{EO}^2 \beta_E^2 + R_{E1}^2 \beta_E^2 + R_f^2) 2 \beta_E^2$$

5.2.3.2 Calculation Sectors and Zones

The cross-sectional grain geometry in motor end sections is based on the geometry at the forward and aft tangent planes. Figures 5.32 and 5.33 define sectors and zones used for burn-surface area and propellant volume calculations. The general forked wagon wheel grain in Figure 5.32 is divided into 13 sectors. Depending on the exact grain geometry, more than one sector may exist in a zone. This is the case for zone A, Figure 5.33 and initial sectors 1, 2, 3, and 4 in Figure 5.32. The opposite may exist where only a portion of a sector is bounded by a zone boundary.

Four zones can exist in the end section. Their boundaries are defined as:

Zone A is that region between the minimum propellant radius, R_1 , and the case opening, R_{E1} . Burning will occur along the side of the propellant tip and on the end face.

Zone B is that region between the case opening, R_{E1} , and where the end-burning surface intersects the case, R_c . Burning will occur along the propellant tip and on the toroidal end face.

Zone C is that region between where the end-burning surface intersects the case, R_c , and the web, $\tau_w - \tau$. Burning will occur only along the side of the propellant tip or in the valley.

Web Zone is that region between the radius $R_f - \tau_w + \tau$ and the radius of the case R_f . Burning will occur along the perimeter of sector 8 and along the end face if the case opening, R_{E1} , is greater than the radius to the web, $R_f - \tau_w + \tau$.

The sectors that are in zones B, C, and web are subdivided into smaller elements. The sectors in zone A are not subdivided. Sector boundaries and zones which the sectors occupy are recalculated as the burning surface regresses.

5.2.3.2 Calculation Sectors and Zones (Continued)

Subroutine ASESUB

Subroutine ASESUB is the control routine to determine the total burning surface area and initial volume of the end sections. The grain cross-section at the adjacent tangent plane is divided into sectors, Figure 5.32. The correct equations from subroutines XRSUB and RASUB are set up and the proper values for the coordinates of the origin of the circular arc, X_{0V} , Y_{0V} , the radius of curvature of the sector, R_T , the angle between the bisector of the propellant tip and the straight side sectors, α_V , and the perimeter length, L_X , are assigned for each sector. After the required parameters have been set for a sector, the sector burning area and volume are determined in subroutine ASESUB.

Subroutine AEPSUB

Subroutine AEPSUB tests for the existence of a sector and is called from subroutine ASESUB. If a sector has burned out or was not present in the initial grain configuration, control is returned to subroutine ASESUB; otherwise, control proceeds to subroutine AESUB. After the sector burning area and volume have been determined in subroutine ASESUB, the sector values are added to the sum of the values for the previous sector and control is returned to subroutine ASESUB.

Subroutine AESUB

Subroutine AESUB determines the total burning surface area and initial volume of the sectors. A test is made at the beginning of each zone to determine if the sector exists in that zone.

The parameters X_{R0} , R_{A0} , and Y_{A0} are associated with the beginning coordinates of a sector perimeter, and the parameters X_{RX} , R_{AX} , Y_{AX} and L_X are associated with the end coordinates of a sector perimeter as shown in Figure 5.32. These parameters are determined by subroutines XRSUB and RASUB from the Pythagorean Theorem. The beginning and end points of a sector may not be the beginning and end points of the area to be computed should a sector exist in more than one zone. The beginning coordinates of an area are then subscripted min and the end coordinates are subscripted max instead of 0 and X, respectively. The perimeter length is then defined as L_R . Figure 5.34 shows an example of the min and max coordinates that are used in the following zone A analysis. When an integration scheme is employed, zones B, C, and Web, the min and max coordinates of each increment are determined to compute sector surface area.

USE FOR TYPEWRITTEN MATERIAL ONLY

5.2.3.2

Calculation Sectors and Zones (Continued)

Subroutine RCSUB

Subroutine RCSUB obtains the radius vector from the motor axis to the intersection of the aft burning surface with the case wall, R_c , Figure 5.33. The intersection is determined from the equation of a straight line when $\tau \leq \tau_{E1}$, and from the equation for the aft-dome configuration (circle or ellipse) when $\tau > \tau_{E1}$, where:

$$\tau_{E1} = [(R_{E2} - R_{E1})^2 - h_{E1}^2]^{1/2}$$

Subroutine ARSSUB

Subroutine ARSSUB determines the chord length, L_{RS} , between the minimum point of a sector and a general point along the perimeter of a sector as shown in Figure 5.35. The required parameters are R_{AO} , X_{RO} , R_A , and X_R .

Subroutine ALRSUB

Subroutine ALRSUB determines the arc length of a sector, L_R , from the minimum point of a sector to a general point along the perimeter as shown in Figure 5.35. The required parameters are L_{RS} and R_T .

Subroutine XRSUB

Subroutine XRSUB determines the X-coordinate of a general point on the perimeter of a sector. A separate equation is used for each sector. The required parameters are R_A and τ .

Subroutine RASUB

Subroutine RASUB determines the length of a radius vector from the motor axis to a general point on the perimeter of a sector. A separate equation is used for each sector. The perimeter length along the sector and the distance burned τ are required.

Subroutine HESUB

Subroutine HESUB determines the length of the trapezoidal elements, h_e , used to determine the incremental volumes and areas, Figure 5.36.

USE FOR TYPEWRI MATERIAL ONLY

5.2.3.2.1 Zone A Calculations

When $R_{A0} < R_{E1}$, the surface area is computed in Zone A. When $R_{A0} \geq R_{E1}$, the analysis proceeds to the next zone and R_{Amax} is set to the smaller value of R_{AX} and R_{E1} .

The following Zone A analysis applies to sectors 1 through 7 and 9 through 13. Figure 5.35 is used as an example to define calculation parameters. It illustrates sector 3.

The burning surface area in zone A is determined from an algebraic composition of simple geometric figures such as shown in Figure 5.35.

The value of γ_T , L_R , and L_{RS} are determined as follows:

$$\gamma_T = 2 \arccos \left(\frac{[(2R_T)^2 - L_{RS}^2]^{1/2}}{2R_T} \right)$$

$$L_R = \gamma_T |R_T|$$

$$L_{RS} = \left\{ \left[R_{Amax}^2 - x_{Rmax}^2 \right]^{1/2} - \left[R_{Amin}^2 - x_{Rmin}^2 \right]^{1/2} + \left(x_{Rmax} - x_{Rmin} \right)^2 \right\}^{1/2}$$

where $R_T = R_3 - r$ for sector 3

A_{T0} is the area between the chord L_{RS} and the circular arc L_R and is determined by subtracting the area of the inscribed triangle from the area of the circular sector, abg:

$$A_{T0} = |R_T| L_R - L_{RS} \left[R_T^2 - \frac{(L_{RS})^2}{2} \right]^{1/2}$$

A_{TT} is equal in magnitude to A_{T0} and is positive if R_T is positive, negative if R_T is negative, and zero if R_T is zero. A_{FF} is the area of the trapezoid, abdf:

$$A_{FF} = \left\{ \left[R_{Amax}^2 - x_{Rmax}^2 \right]^{1/2} - \left[R_{Amin}^2 - x_{Rmin}^2 \right]^{1/2} \right\} \frac{(x_{Rmax} - x_{Rmin})}{2}$$

5.2.3.2.1 Zone A Calculations (Continued)

A_R is the area bcd determined by subtracting the triangular area obd from the circular sector obc:

$$A_R = \frac{R_{Amax}^2 \gamma_R - X_{Rmax} [R_{Amax}^2 - X_{Rmax}^2]^{1/2}}{2}$$

where

$$\gamma_R = \arcsin \left(\frac{X_{Rmax}}{R_{Amax}} \right)$$

A_{RO} is the area aef determined by subtracting the triangular area oaf from the area of circular sector oae:

$$A_{RO} = \frac{R_{Amin}^2 \gamma_{RO} - X_{Rmin} [R_{Amin}^2 - X_{Rmin}^2]^{1/2}}{2}$$

where

$$\gamma_{RO} = \arcsin \left(\frac{X_{Rmin}}{R_{Amin}} \right)$$

The burning area along the side of the propellant tip is $h_E(L_R)$ and the total sector area A_{EE} is:

$$A_{EE} = 2 NO [L_R h_E + A_{TT} + A_{FF} + A_R - A_{RO}]$$

The sector volume is:

$$DV = (A_{FF} + A_R - A_{RO} + A_{TT}) h_E NO 2$$

If the maximum point of the sector, R_{Amax} is within zone A, then control is returned to subroutine AEPSUB; otherwise, computation will proceed to zone B.

5.2.3.2.2 Zone B Calculations

When R_{AO} for a sector is less than R_C , surface area is computed in zone B. When R_{AO} is greater than or equal to R_C , computation proceeds to Zone C.

Initially, R_{Amax} is set to the smallest value of R_{AX} and R_C from which L_{Rmax} is determined by subroutine ARSSUB and ALRSUB. L_{R1}

USE FOR TYPEWRI. IN MATERIAL ONLY

5.2.3.2.2 Zone B Calculations (Continued)

Is determined in the same manner from R_{Amin} . The burning surface area is then computed in increments of ΔL where $\Delta L = R_f$ (DLRF) by integrating over the perimeter length, L_{Rmax} . DLRF is an input parameter. For each increment, R_{Amax} is determined from $L = L_{R1} + \Delta L$, and R_{Amin} is determined from $L = L_{R1}$, Figure 5.34. The X-coordinate, X_R , of the centroid, Figure 5.36, is obtained from subroutine XRSUB and the radius vector R_A is obtained from subroutine RASUB. The angle γ_R is:

$$\gamma_R = \arcsin \left(\frac{X_R}{R_A} \right)$$

The cross-sectional area of the end face is equal to the product of the incremental arc length $(\lambda - \lambda_{min})\tau$, and the arc length, $\gamma_R R_A$, through which the centroid is rotated.

The surface area along the side of the propellant tips are approximated by trapezoidal increments and is added to the area of the end face. The burning surface areas and volumes of each increment are added to the sum of the previous increment values. They are determined for an increment as follows:

$$A_{EE} = 2 NO \left[(L - L_{R1}) \frac{h_E + h_E^I}{2} + R_A \gamma_R (\lambda - \lambda_{min})\tau \right]$$

where

$$h_E = |h_{ER} - [\gamma^2 - (R_{Amax} - R_{E1})^2]^{1/2}|$$

$$h_E^I = |h_{ER} - [\gamma^2 - (R_{Amin} - R_{E1})^2]^{1/2}|$$

$$DV = 2 NO (R_{Amax} - R_{Amin}) \frac{h_E + h_E^I}{2} \gamma_R R_A$$

The next increment in the sector is determined by setting $L_{R1} = L$, $h_E^I = h_E$, $R_{Amin} = R_{Amax}$, and $L_{R1} = L_{R1} + \Delta L = L$. L is set to L_{Rmax} for the last iteration for a sector.

USE FOR TYPEWRITTEN MATERIAL ONLY

5.2.3.2.2 Zone B Calculations (Continued)

The parameters R_A , R_{Amax} , L , and h_E are determined for each increment.

The toroidal end burning area is formed by revolving an arc about the motor axis, and is equal to the product of the length of the arc increment and the arc length through which the centroid of the arc increment is rotated (Theorem of Pappus). The area along the side of the propellant tip is determined from the trapezoids.

The radial vector, R_A , to the centroid of the arc increment, as shown in Figure 5.36, is:

$$R_A = \frac{2\tau}{\lambda - \lambda_{min}} \sin\left(\frac{\lambda - \lambda_{min}}{2}\right) \sin\left(\frac{\lambda + \lambda_{min}}{2}\right) + R_{E1}$$

where $\frac{2\tau}{\lambda - \lambda_{min}} \sin\left(\frac{\lambda - \lambda_{min}}{2}\right)$ is the distance from the centroid of the arc increment to the origin of the circle about which the arc increment is revolved.

The angles λ and λ_{min} are determined as follows:

$$\lambda = \arcsin\left(\frac{R_{Amax} - R_{E1}}{\tau}\right)$$

$$\lambda_{min} = \arcsin\left(\frac{R_{Amin} - R_{E1}}{\tau}\right)$$

If R_{AX} is less than R_C , computation will proceed to the next sector. If R_{AX} is greater than or equal to R_C , computation will proceed to zone C.

5.2.3.2.3 Zone C Calculations

The burning surface area in Zone C consists only of the surface along the side of the propellant tip, and is determined from trapezoidal elements. If a sector exists in zones B and C, R_A is set equal to R_{Amax} of zone B; otherwise, $R_A = R_{A0}$. With R_A , L_{R1} is determined from subroutines ARSSUB and ALRSUB. The length of the initial edge, h_E^1 , of the trapezoidal element is determined in subroutine HESUB as follows:

5.2.3.2.3 Zone C Calculations (Continued)

If $R_A \leq R_{E2}$,

$$h_E^i = h_{ER} - h_{E1} \left(\frac{R_A - R_{E1}}{R_{E2} - R_{E1}} \right)$$

Or if $R_A > R_{E2}$,

$$h_E^i = h_{ER} + \frac{(R_f^2 - R_A^2)^{1/2}}{\beta_E} - h_{E0}$$

An incremental length ΔL is added to L_{R1} and a corresponding R_A is determined from subroutine RASUB. The length of the top edge h_E of the trapezoidal element is determined from the above equations, and the elemental trapezoidal area and volume are added to the previous sectors as follows:

$$A_{EE} = 2 NO (L - L_{R1}) \left(\frac{h_E + h_E^i}{2} \right)$$

where

$$R_A = \frac{R_{Amax} + R_{Amin}}{2}$$

$$DV = 2 NO (R_{Amax} - R_{Amin}) \frac{(h_E + h_E^i)}{2} \gamma_R$$

5.2.3.2.4 Web Zone Calculations

Subroutine AWESUB determines burning surface area and initial volume of the web zone.

The burning surface area of sector 8 is determined first from the trapezoidal element of length h_E , which is determined from $R_{A0} = R_f - \tau_w + \tau$, as follows:

$$AWE = 2 NO h_E L_8$$

L_8 is the perimeter length of sector 8.

The volume of propellant in the web zone is determined as follows: (geometric symbols are given in Figures 5.33 and 5.36)

5.2.3.2.4 Web Zone Calculations (Continued)

If $R_{E2} > (R_f - \tau_w + \tau)$, calculate the parameter,

$$h_{EFC} = \frac{(R_{E2} - R_f + \tau_w - \tau)}{R_{E2} - R_{E1}} h_E'$$

and the volume as:

$$\begin{aligned} DV = & \left\{ \left[\left(\frac{R_f}{\beta_E} \right)^2 h_{E2} - \frac{h_{E2}^3}{3} \right] \beta_E^2 + (h_{ER} - h_{EO}) R_f^2 \right. \\ & + \left[(R_f - \tau_w + \tau)^2 + R_{E2} (R_f - \tau_w + \tau) + R_{E2}^2 \right] \frac{h_{EFC}}{3} \\ & \left. - (h_{EFC} + h_{ER} - h_{E1}) (R_f - \tau_w + \tau)^2 \right\} \pi \end{aligned}$$

If $R_{E2} \leq (R_f - \tau_w)$, calculate the parameter,

$$Z_1 = \frac{[R_f^2 - (R_f - \tau_w + \tau)^2]^{1/2}}{\beta_E}$$

and the volume as:

$$\begin{aligned} DV = & \left\{ \left[\left(\frac{R_f}{\beta_E} \right)^2 Z_1 - \frac{Z_1^3}{3} \right] \beta_E^2 + (h_{ER} - h_{EO}) R_f^2 \right. \\ & \left. - (Z_1 + h_{ER} - h_{EO}) (R_f - \tau_w + \tau)^2 \right\} \pi \end{aligned}$$

The additional end burning area and initial propellant volume, when $R_{E1} > R_f - \tau_w + \tau$, are determined as follows:

When $R_{E1} < R_C$

$$A_{WE} = 2 NO \theta_1 (R_{E1}^2 - R_{AO}^2)$$

$$DV = \pi h_E (R_{E1}^2 - R_{AO}^2)$$

The toroidal end burning surface exists when $R_C > R_{E1}$. The re-

5.2.3.2.4 Web Zone Calculations (Continued)

volved area is a product of the arc length and the circumference of the circle described by the centroid of the arc length. The radial vector to the centroid of the arc increment, Figure 5.36, is:

$$R_A = \frac{2\tau}{\lambda - \lambda_{\min}} \sin\left(\frac{\lambda - \lambda_{\min}}{2}\right) \sin\left(\frac{\lambda + \lambda_{\min}}{2}\right) + R_{E1}$$

The toroidal surface area is determined from the product of the arc length, $(\lambda - \lambda_{\min})\tau$, and the circumference of the circle, $2\pi R_A$, described by the centroid of the curve:

$$A_{WE} = 2(\lambda - \lambda_{\min}) \tau \pi R_A$$

Pitch and roll moments of Inertia (MOI) and the center of gravity (CG) location during motor burning can be calculated. The roll moment (J-ROLL) is taken about the longitudinal axis of the motor. The pitch moment (J-X-Y) is taken about an axis passing through the motor CG and centerline. The center of gravity is measured from the aft tangent plane. The value is positive when the CG is forward of the aft tangent plane. The moments about the pitch and yaw axes are assumed to be equal. This assumption is valid for any configuration with an even multiple of 4 propellant tips.

The MOI and CG location of the motor are based on the combined values of each section; fore-head, cylinder, and aft-head.

The pitch MOI of the fore-head is initially determined about the forward tangent plane and then transferred to the aft tangent plane, the cylindrical and aft-head section MOI's are initially determined about the aft tangent plane. These values about the aft-tangent plane are then transferred to the motor CG. The transfer formula is:

$$J_{CG} = J_{\text{aft tangent plane}} + d^2 W/g_0$$

where d is the distance between the motor CG and aft tangent plane.

The MOI of a body with respect to a given axis is defined as the product of the mass and the square of the distance from the axis. If $dm = dW/g_0$ represents an elemental mass and Y its distance from an axis, the MOI, J , of the object about this axis will be equal to $\int Y^2 dW/g_0$, Reference 8.

The CG is that point at which the mass of an object is concentrated so that the moment of the concentrated mass about any axis or plane is the same as the sum of the moments of all the elements of the mass about the same axis or plane. The sum of the moments from a plane is $\int \frac{x dW}{g_0}$

and the CG is defined as: $\bar{x} = \frac{\int x dW}{W}$

where

\bar{x} = distance to the CG location from a plane.

The MOI and CG location for each section are determined from a summation of incremental volumes and areas about the desired axis. The incremental volumes are hollow circular cylinders for the cylindrical section and thin shells for the head-end sections. The radius of gyration for the cylindrical section is determined from the roll MOI of the grain cross-sectional area. The incremental rectangular MOI's are first taken about the CG of the incremental volume and then transferred to the desired axis. The transfer formula is $I = I_{CG} + d^2 m$, where I_{CG} is the MOI about the CG and d is the distance to the reference axis.

5.3.1

Roll Moment of Inertia

The roll MOI of an incremental thin shell for the head sections are determined from the fundamental equation; Figure 5.37:

$$J_P = \frac{W}{g_0} r^2, \text{ slug-in}^2$$

The thin shells are based on a subdivision of sector boundaries.

The roll MOI of the cross sectional area of an incremental hollow circular cylinder, Figure 5.38, is determined from the fundamental equation:

$$J_P = \frac{(\theta_{r1} + \theta_{ro})}{4} (r_o^4 - r_1^4) \text{ NO, in}^4$$

where

r_o = outer incremental radius, in

r_1 = inner incremental radius, in

The radius of gyration is determined from:

$$K = \left(\frac{J_P}{AFP} \right)^{1/2}, \text{ in}$$

where AFP is the cross sectional area of propellant, in².

5.3.1.1

Motor End Sections

The roll MOI's for the motor end sections are determined in sub-routines PT1AA and SD1D13 from a summation of thin shells of each sector and the web region as follows:

$$AJPP = \sum_1^{13} [J_{PX}] + (R_{AO}^2 + R_A^2) (R_A^2 - R_{AO}^2) \pi \frac{(h_E + h'_E)}{2} \frac{\rho_f}{2g_0}$$

$$\text{where, } J_{PX} = (R_A^2 + R_{AO}^2) \frac{W_1}{2g_0}$$

$$W_1 = (\theta_{r1} + \theta_{ro}) \text{ NO } \rho_f (R_A^2 - R_{AO}^2) \frac{(h_E + h'_E)}{4}$$

R_{AO} is the minimum radius of the increment

R_A is the maximum radius of the increment

5.3.1.2

Cylindrical Section

The roll MOI for the cylindrical section is determined in subroutine SEGSUB using the radius of gyration of the propellant cross section. The radius of gyration of the cross section is determined in subroutines PT1AA and SD1D13 from a summation of the roll MOI for the elementary hollow circular cylinders of each sector and the web region as follows:

$$J_{PP} = \sum_1^{13} \left[\frac{(\theta_{r1} + \theta_{ro})}{4} (R_A^4 - R_{AO}^4) NO \right] + \frac{\pi}{2} (R_A^4 - R_{AO}^4)$$

$$W_T = \sum_1^{13} \left[\frac{(\theta_{r1} + \theta_{ro})}{2} (R_A^2 - R_{AO}^2) NO \right] + \pi (R_A^2 - R_{AO}^2)$$

$$K_{GY} = \left[\frac{J_{PP}}{W_T} \right]^{1/2}$$

The radius of gyration is calculated only at the input reference planes and linearly interpolated at each increment dividing plane. The cylindrical section total roll MOI is determined from a summation of individual mass addition region roll MOI values as follows, Figure 4.3:

$$I_{PCYL} = \sum_1^{NI} \left[(\pi R_{FHI}^2 - A_{PHI}) K_{GYHI}^2 + (\pi R_f^2 - A_P) K_{GY}^2 \right] \frac{\Delta Z \rho_f}{2 g_o}$$

The subscripted HI parameters are the values at the adjacent upstream increment dividing plane. NI is the total number of increment dividing planes.

5.3.2

Pitch MOI

The pitch MOI's of the head sections are taken with reference to the adjacent tangent plane such that:

$$J_B = \frac{W}{g_o} \left(\frac{r^2}{2} + \frac{L^2}{12} \right) + \frac{W}{g_o} R_{CG}^2$$

where

r = radius to CG of cross section, in

L = incremental length of shell, in

R_{CG} = distance from reference axes (tangent plane) to the CG, in

5.3.2

Pitch MOI (Continued)

For the incremental hollow circular cylinders of the cylindrical section the calculations are taken with reference to the CG of the incremental cylinder and then transferred to the aft tangent plane such that:

$$J_B = \frac{W}{g_o} \left(\frac{K^2}{2} + \frac{L^2}{12} \right) + \frac{W}{g_o} d^2$$

where

K = radius of gyration of cross section, in²

d = distance from the CG to aft tangent plane, in

5.3.2.1

Motor End Sections

The pitch MOI for the motor end sections are determined in sub-routines PT1AA and SD1D13 from a summation of MOI's for elemental thin shells of each sector and the web region as follows:

$$AJBB = \sum_1^{13} \left\{ \left[\frac{(h_E + h'_E)^2}{2} \right] \frac{W_I}{12g_o} + \frac{J_{PX}}{2} + \frac{W_I R_{CG}^2}{g_o} \right\} \\ + \frac{R_{CG}^2 W_I}{g_o} + \left[\frac{(h_E + h'_E)^2}{12} + (R_A^2 + R_{AO}^2) \right] \frac{W_I}{4g_o}$$

where

$$R_{CG} = \begin{cases} h_{EO} + \frac{(h_E + h'_E)}{4} & \text{for the fore-head and} \\ \frac{h_E + h'_E}{4} & \text{for the aft-head} \end{cases}$$

$$J_{PX} = (R_A^2 + R_{AO}^2) \frac{W_I}{2g_o}$$

5.3.2.2

Cylindrical Section

The MOI is determined in subroutine SEGSUB from a summation of individual mass addition region values as follows:

$$I_{BCYL} = \sum_1^{NI} \left\{ \frac{(\pi R_{FHI}^2 - A_{PHI})}{2} \left[K_{GYHI}^2 + 2 \left(\frac{3}{4} (A_{INCW} - A_{INCHI}) + h_{CO} - A_{INCW} \right)^2 \right] \right\}$$

5.3.2.2 Cylindrical Section (Continued)

$$+ \frac{2(AINCW-AINCHI)^2}{12}] + [K_{GY}^2 + 2 \left(\frac{AINCW-AINCHI}{4} + h_{CO} - AINCW \right)^2]$$

$$\left. \frac{(\pi R_f^2 - A_P)}{2} \right\} \frac{\rho_f (AINCW-AINCHI)}{2 g_o}$$

5.3.3 CG of Sections

The CG for each section is determined from a summation of the moments of incremental volumes about the desired axis. The CG for the cylindrical section is determined from a summation of the moments of incremental hollow circular cylinders about the aft tangent plane, Figure 5.38, and the CG for the head sections are determined from a summation of the moments of thin shells about the adjacent tangent plane, Figure 5.37. Thus:

$$\overline{MX} = \frac{W}{g_o} \overline{X} = \sum X_i \frac{dW}{g_o}$$

therefore

$$\overline{X} = \sum X_i \frac{dW}{W}$$

where

X_i = moment arm of Incremental volume, in

dW = Incremental volume weight, lb

W = total weight, lb

5.3.3.1 Head Sections

The CG is determined in subroutines PT1AA and SD1D13 from a summation of moments for elemental thin shells of each sector and the web region as follows:

$$\overline{X}_i = \sum_1^{13} [(W_T \overline{X}_i \text{ previous} + (h_E + h'_E) \frac{W_i}{4}) / (W_i + W_T)]$$

$$\text{where } W_T = \sum_1^{13} W_i$$

subscript i is N for aft-head,

H for fore-head.

5.3.3.2

Cylindrical Section CG Location

The CG location is determined in subroutine SEGSUB from a summation of the individual mass addition region moments as follows, Figure 5.38:

$$AOMCYL = \sum_1^{NI} \left\{ \left[\left(\frac{3(AINCW-AINCHI)}{4} + h_{CO} - AINCW \right) (\pi R_{fHI}^2 - A_{PHI}) + \left(\frac{AINCW-AINCHI}{4} + h_{CO} - AINCW \right) (\pi R_f^2 - A_p) \right] \frac{(AINCW-AINCHI)}{2} \rho_f \right\}$$

5.3.4

Motor MOI and CG

Motor Roll MOI:

$$J_{PP} = J_{PHed} + J_{PNoz} + I_{Pcyl}$$

The motor CG location and pitch MOI are determined by a transfer of axes of the MOI of each section as follows:

Motor CG:

$$\bar{X}_{IH} = \frac{(\bar{X}_H + h_{CO}) V_{FH} \rho_f + AOMCYL - \bar{X}_N V_{FN} \rho_f}{W_f}$$

Fore-Head Section Pitch MOI:

$$J_{BHed} = (J_{BHed})_{Aft} - \left[\bar{X}_H^2 - (\bar{X}_H + h_{CO} - \bar{X}_{IH})^2 \right] \frac{V_{FH} \rho_f}{g_0}$$

Aft-Head Section Pitch MOI:

$$J_{BNoz} = (J_{BNoz})_{Aft} - \left[\bar{X}_N^2 - (\bar{X}_N + \bar{X}_{IH})^2 \right] \frac{V_{FN} \rho_f}{g_0}$$

Cylindrical Section Pitch MOI:

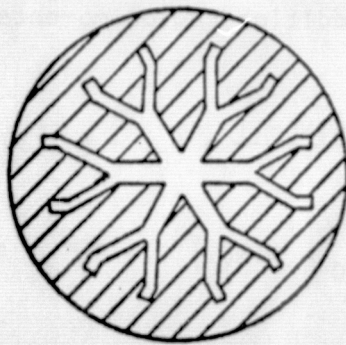
$$I_{Bcyl} = (I_{Bcyl})_{Aft} - \left\{ \left(\left[\bar{X}_{IH} W_f - (\bar{X}_H + h_{CO}) V_{FH} \rho_f + \bar{X}_N V_{FN} \rho_f \right] / \left[W_f - (V_{FH} + V_{FN}) \rho_f \right] \right)^2 \right\} / g_0$$

Motor Pitch MOI:

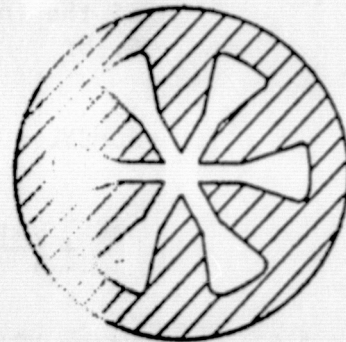
$$J_{BB} = J_{BHed} + J_{BNoz} + I_{Bcyl}$$

USE FOR TYPEWRITTEN MATERIAL ONLY

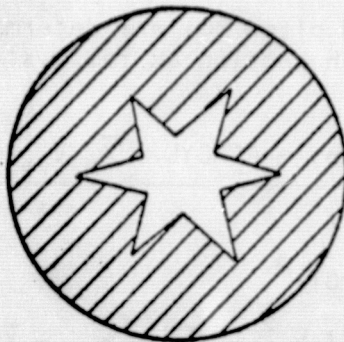
ORIGINAL PAGE IS
OF POOR QUALITY



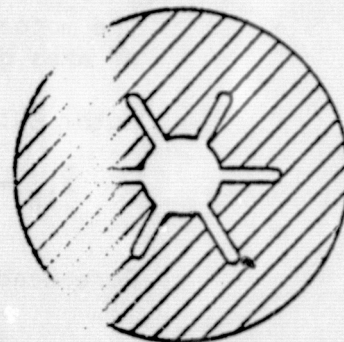
Forked Wagon Wheel



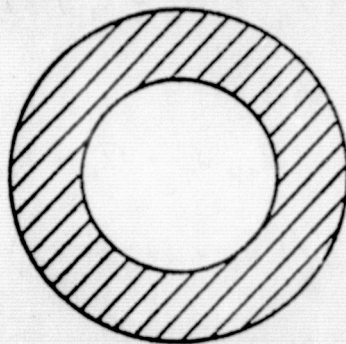
Conventional Wagon Wheel



Standard Star

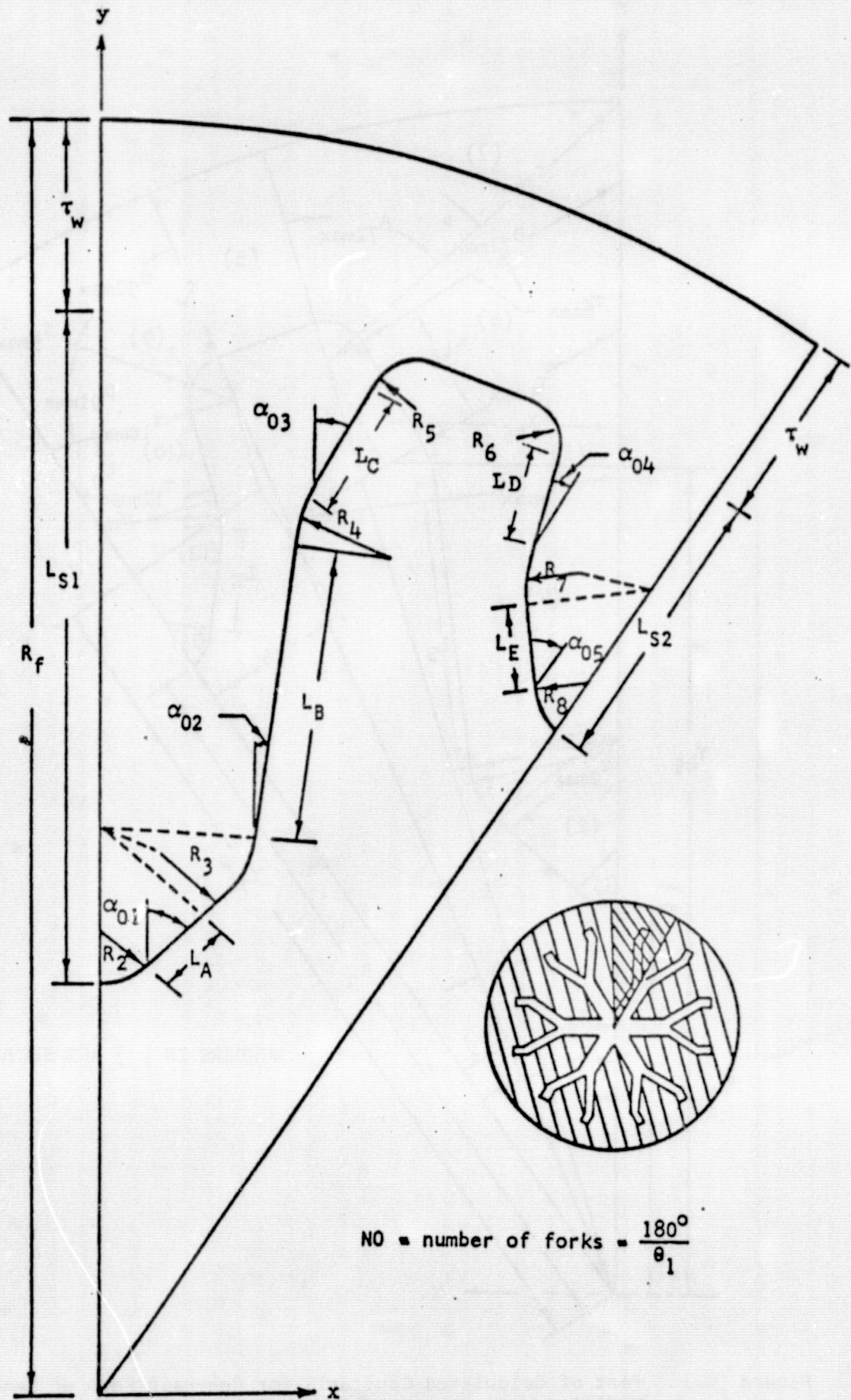


Dotted-Cone



Circular Port

Figure 5.1. Grain Design Patterns



$$N_0 = \text{number of forks} = \frac{180^\circ}{\theta_1}$$

Figure 5.2 One-Half Fork of General Modified Wagon Wheel Configuration

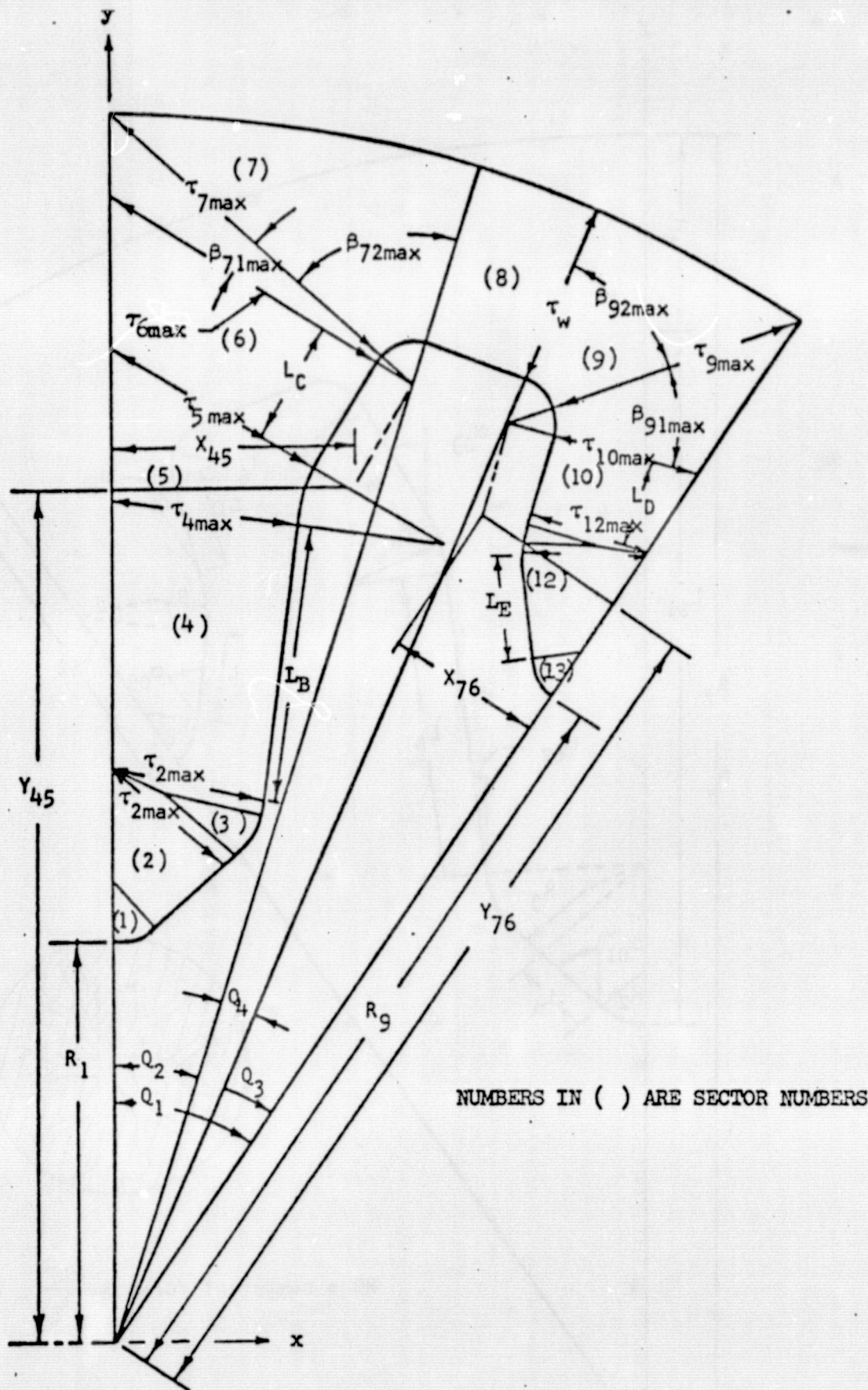


Figure 5.3 Part of Calculated Constants for One-Half Fork of General Modified Wagon Wheel Configuration Produced by PLNCNS Sub-routine

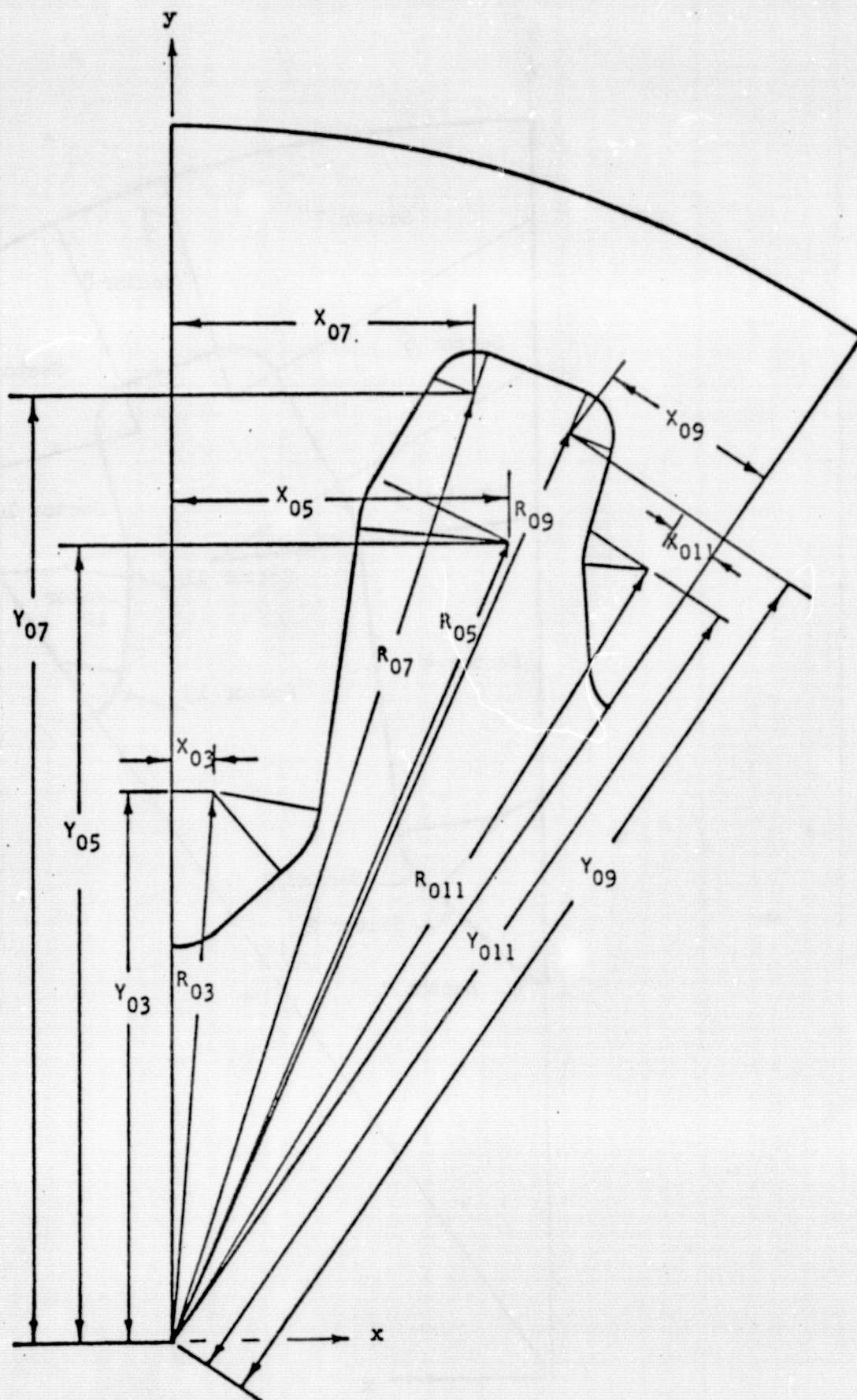


Figure 5.4 Part of Calculated Constants for One-Half Fork of General Modified Wagon Wheel Configuration Produced by PLNCNS Sub-routine

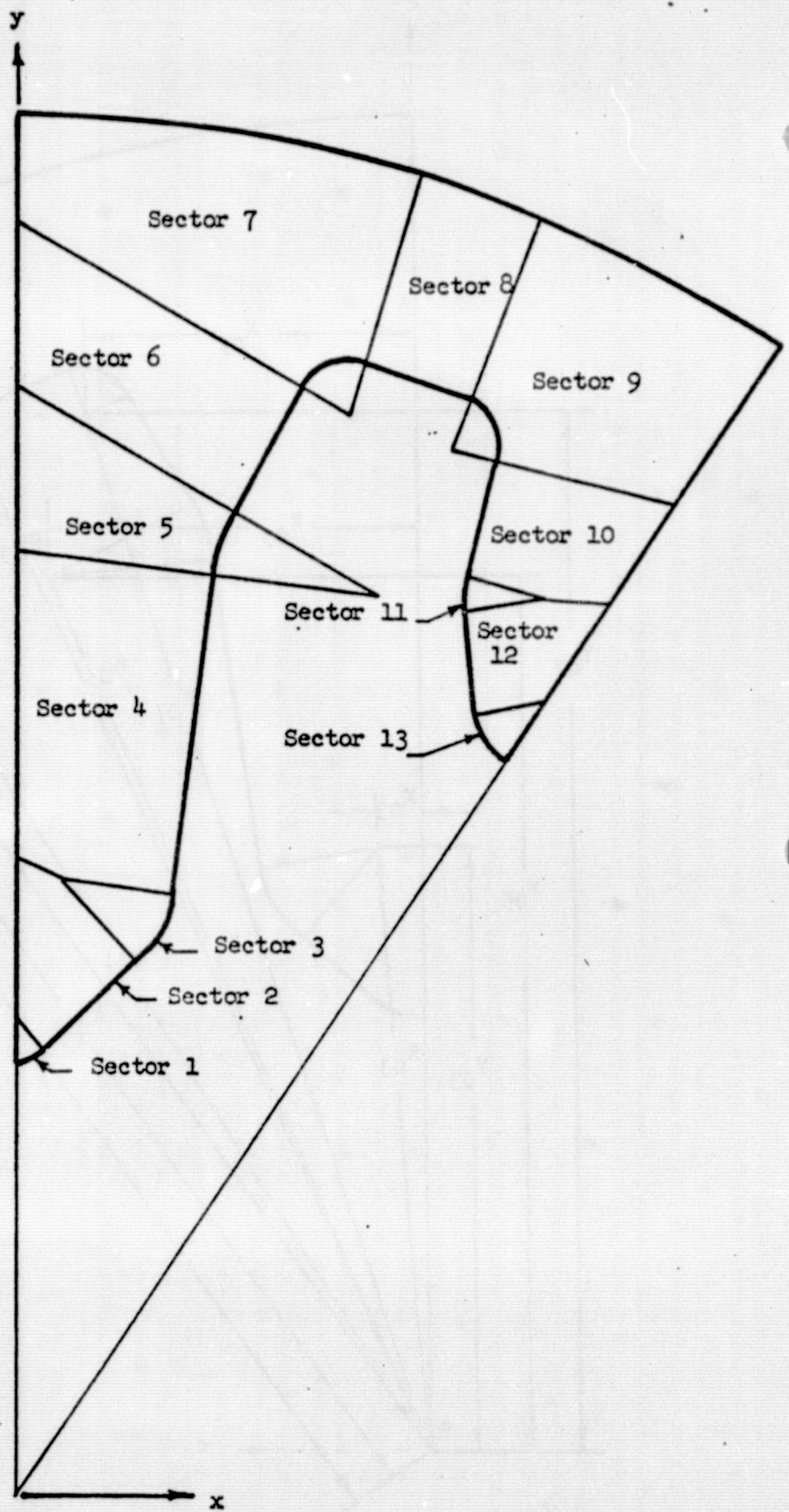


Figure 5.5 Sector Definition

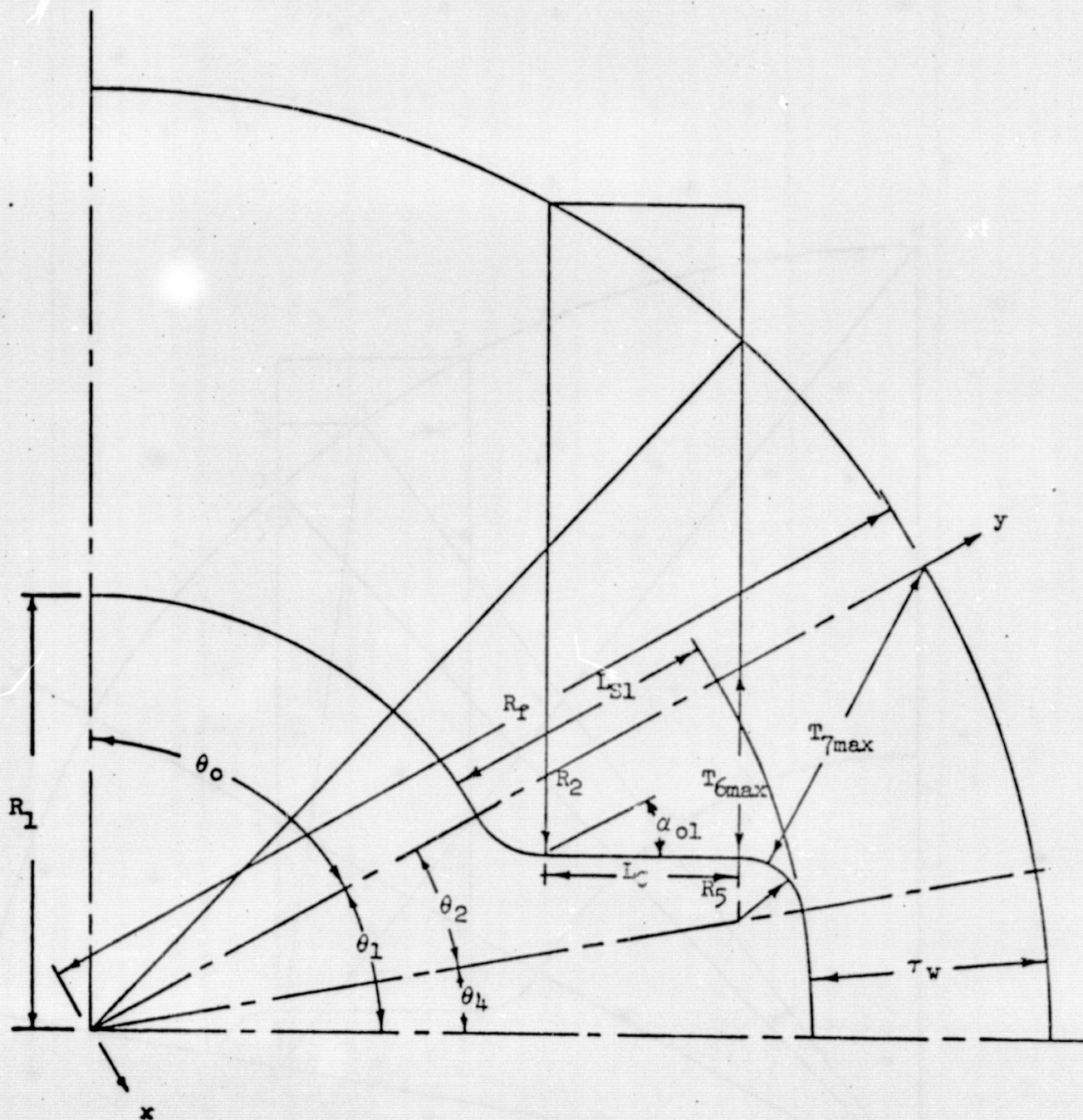


Figure 5.6 Slotted-Cone

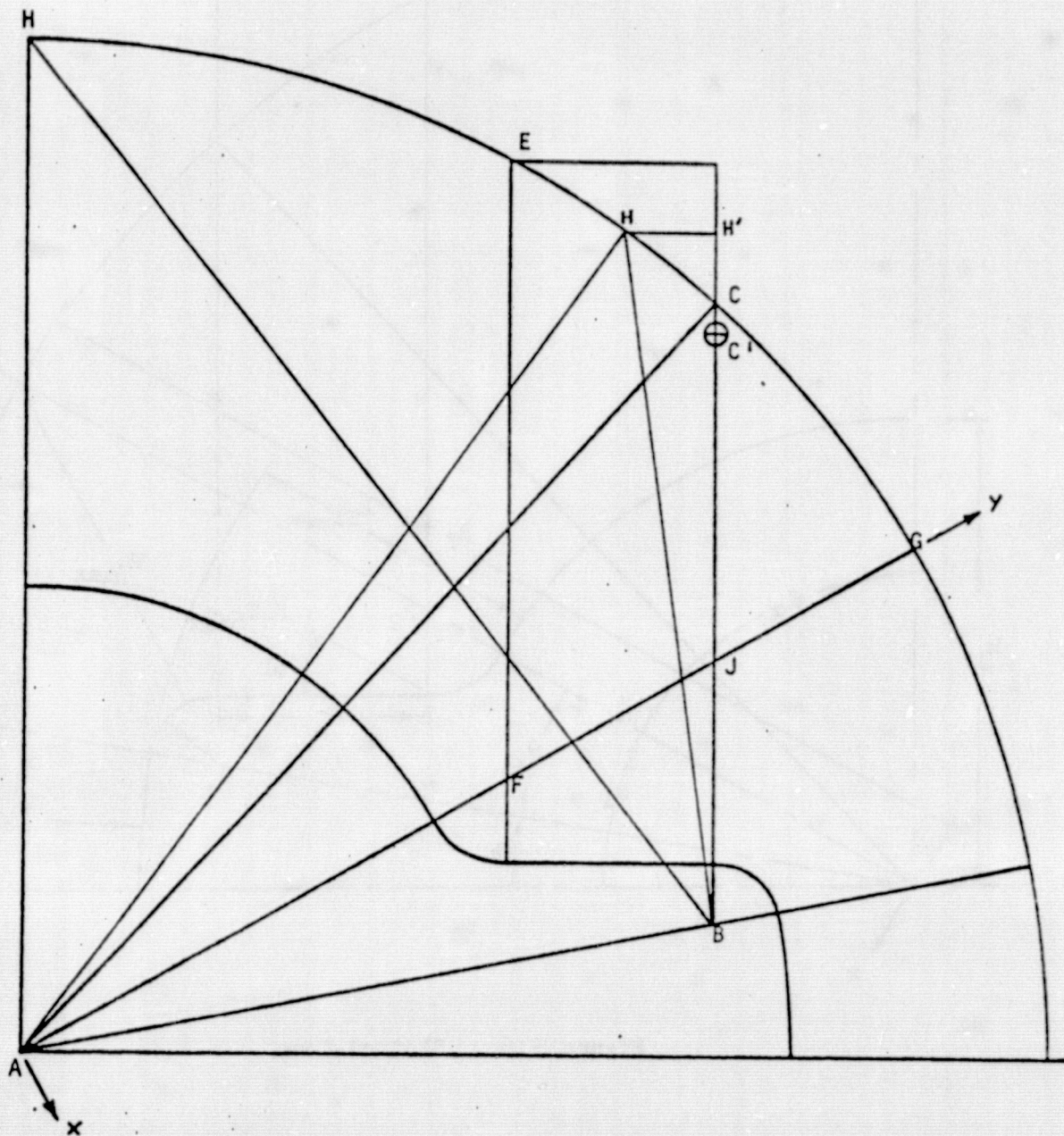
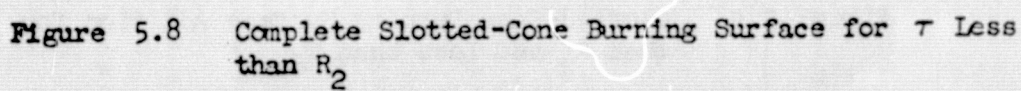
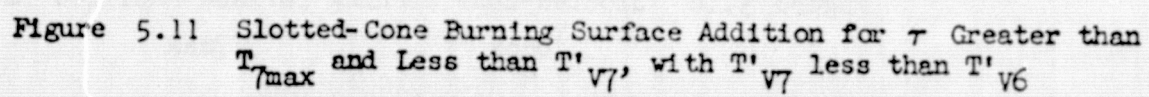
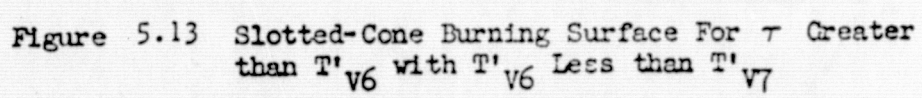


Figure 5.7 Slotted-Cone Addition to Standard Star Showing Location of Fixed Geometry Points



119





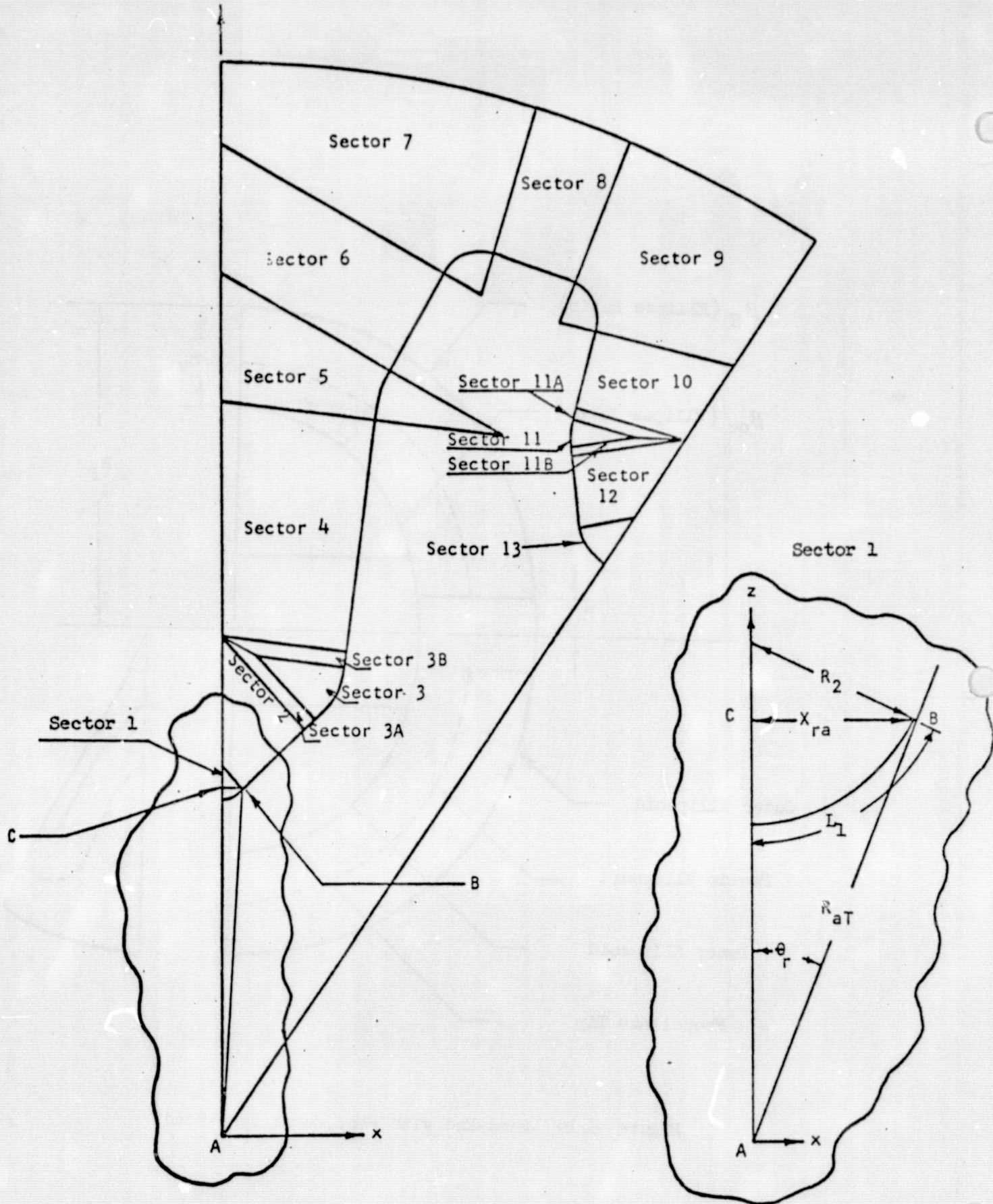


Figure 5.15 Sectors for Block No. 1

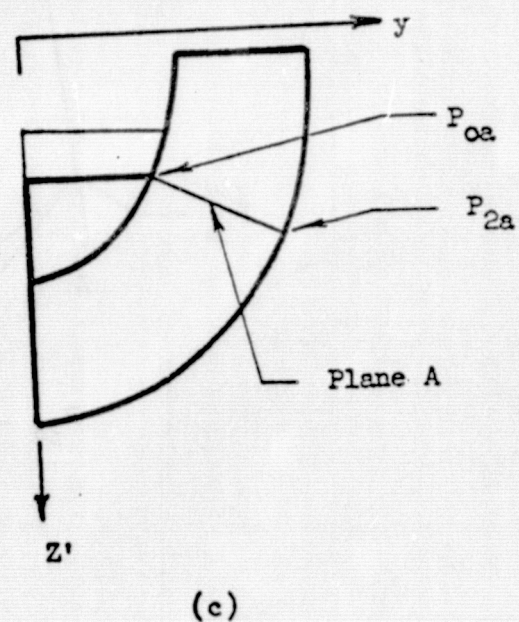
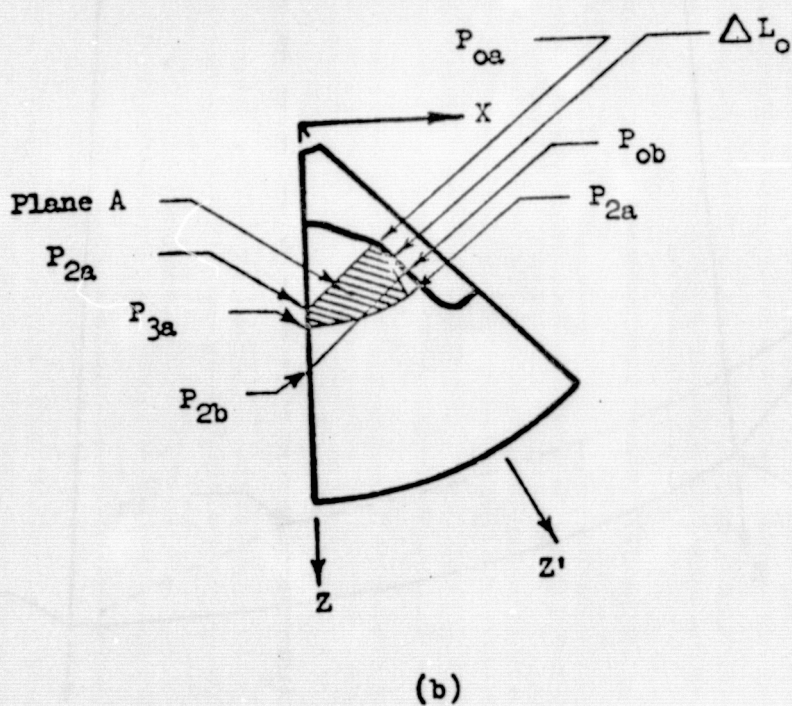
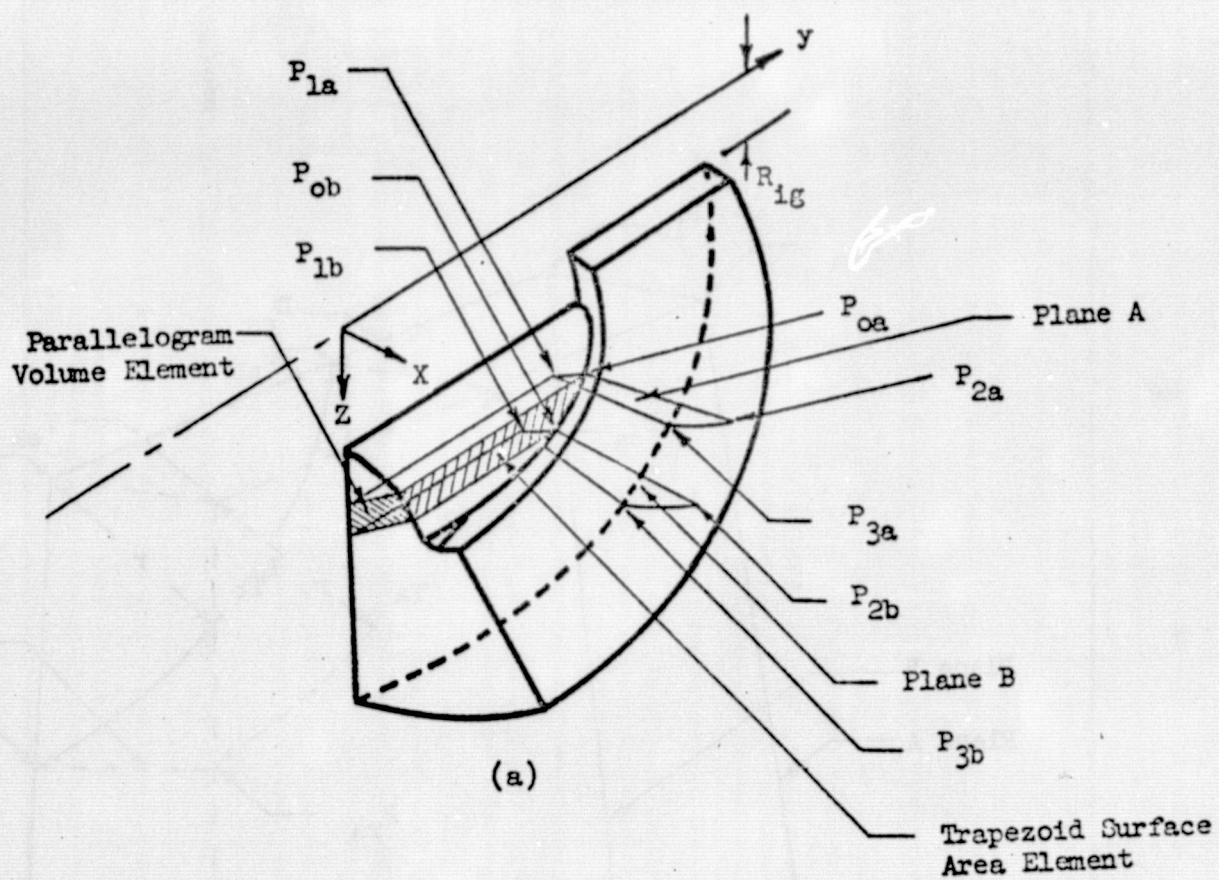
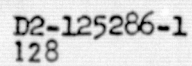


Figure 5.16 Head-End with Web Plane Definition

D2-125286-1



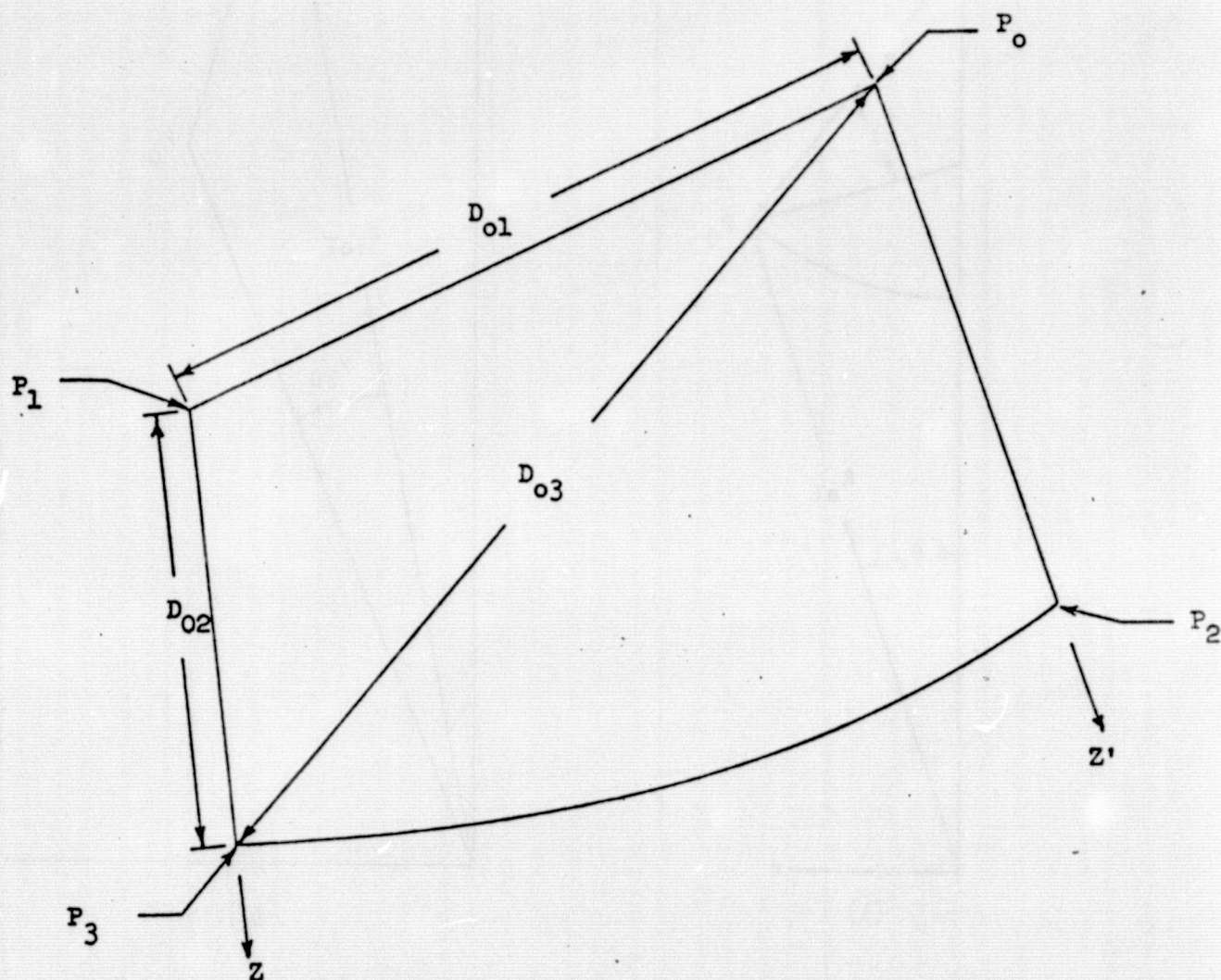


Figure 5.18 Plane for Block 1 Analysis

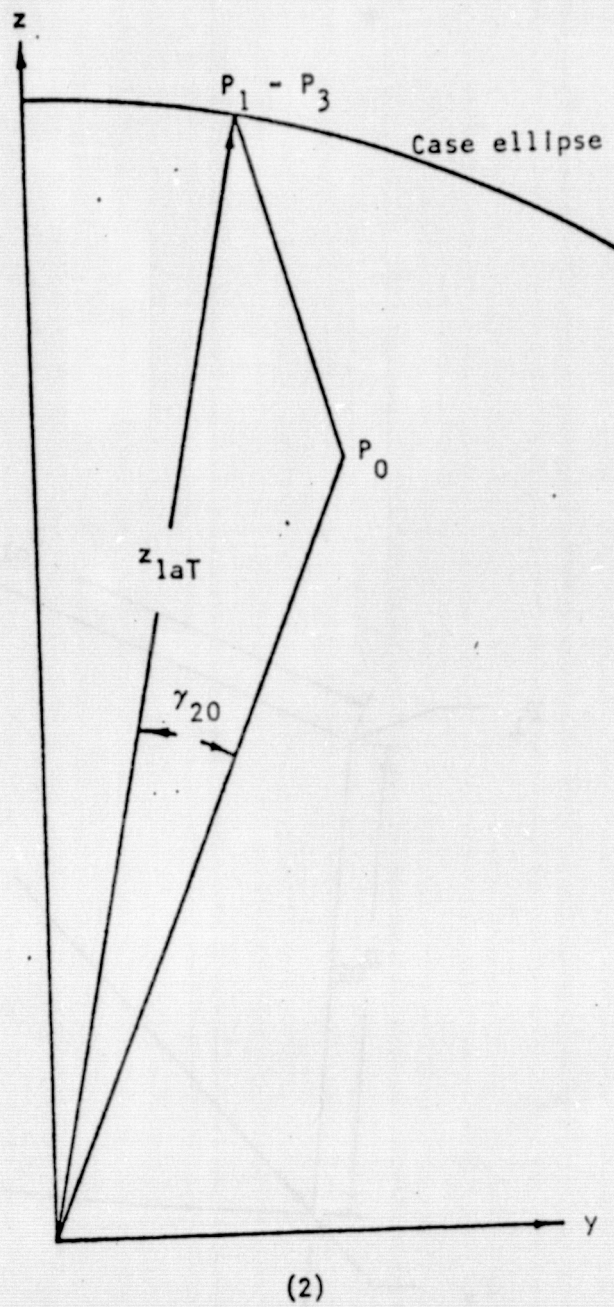
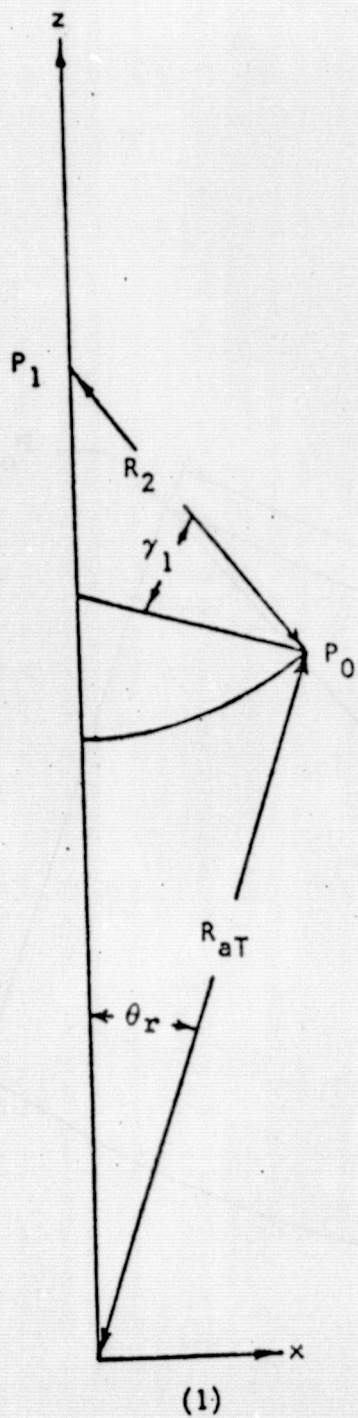


Figure 5.19 γ_1 for Subroutine GAMSUB

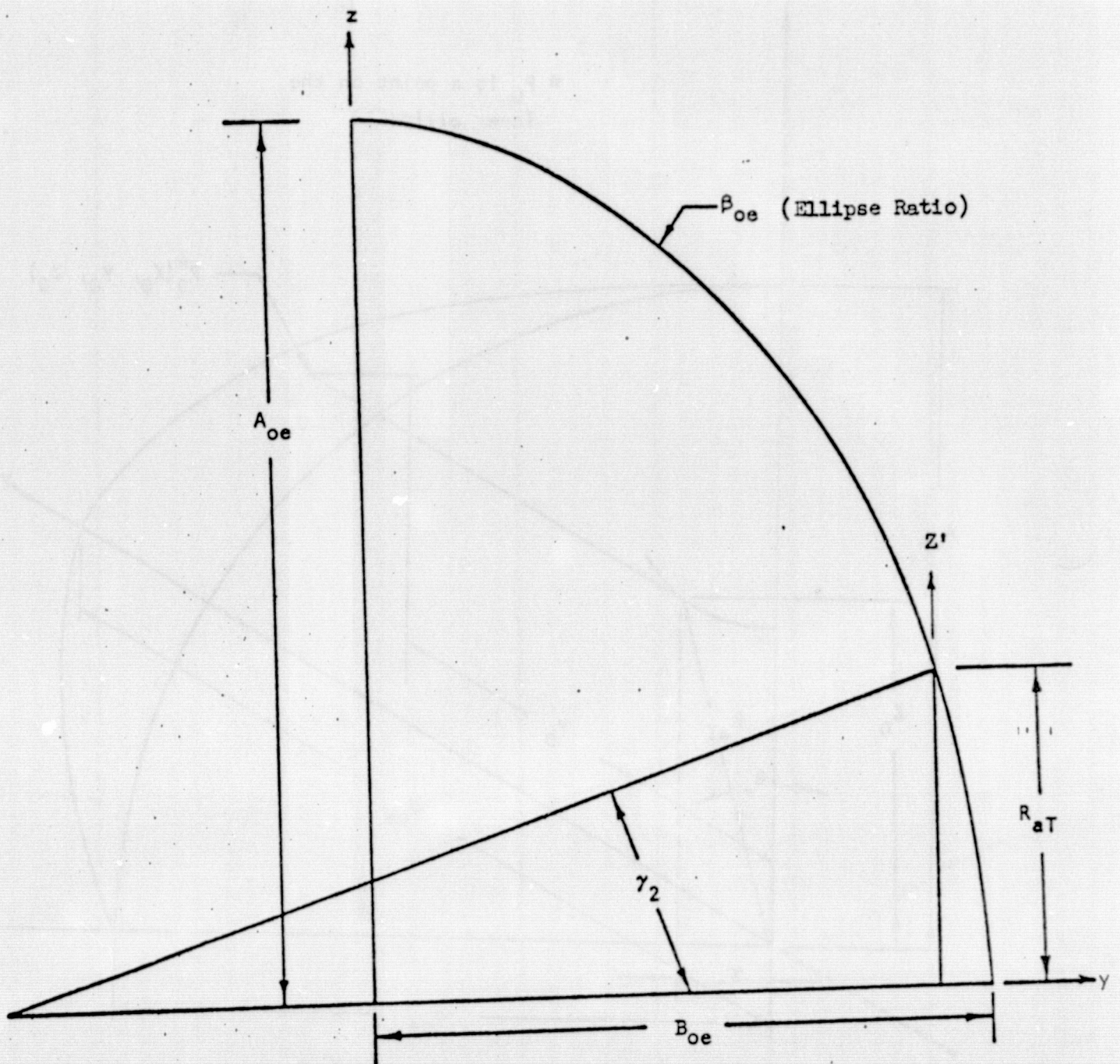


Figure 5.20 γ_2 for Subroutine GAM2S

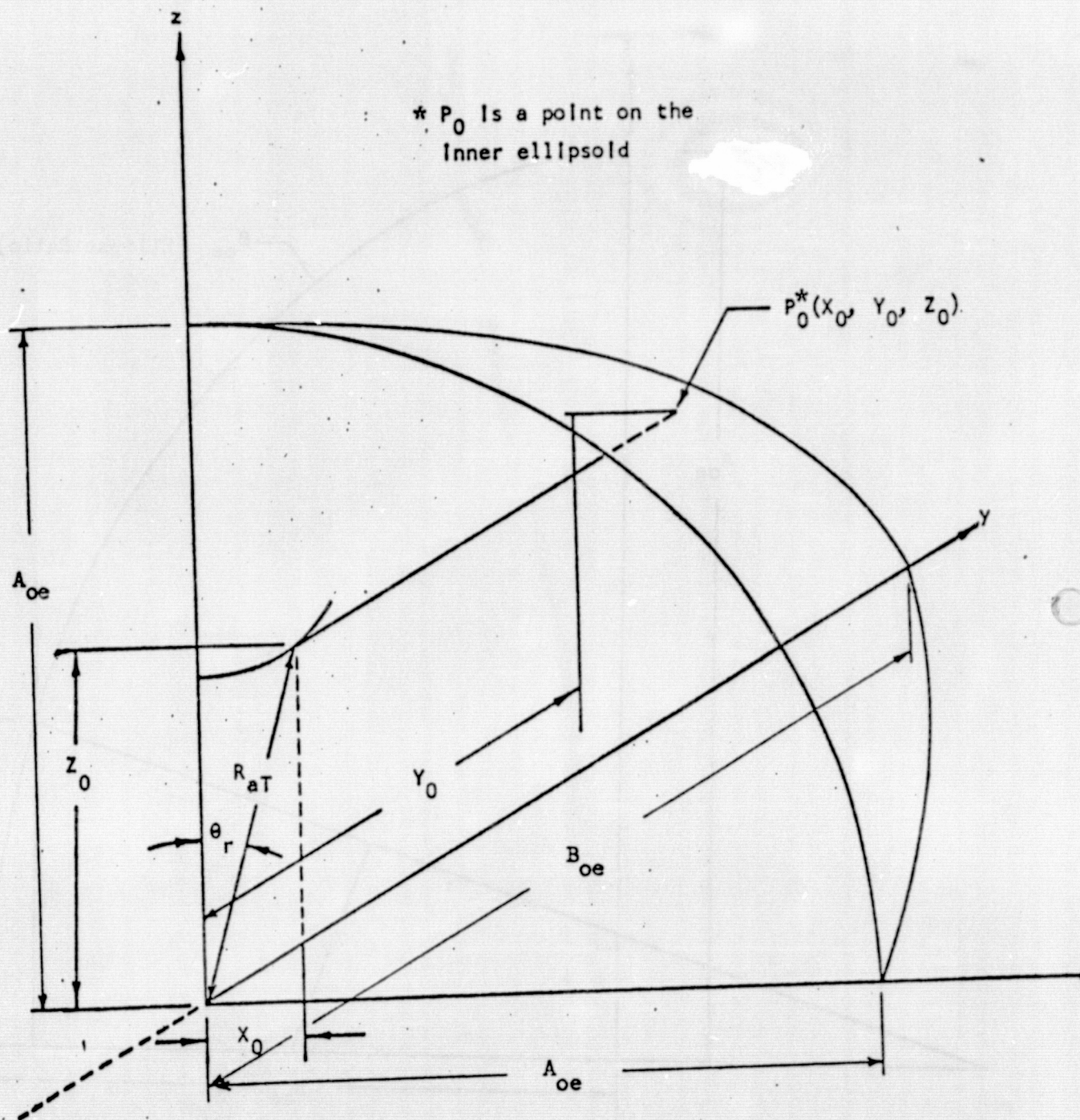


Figure 5.21 P_0 for Subroutine POSUB

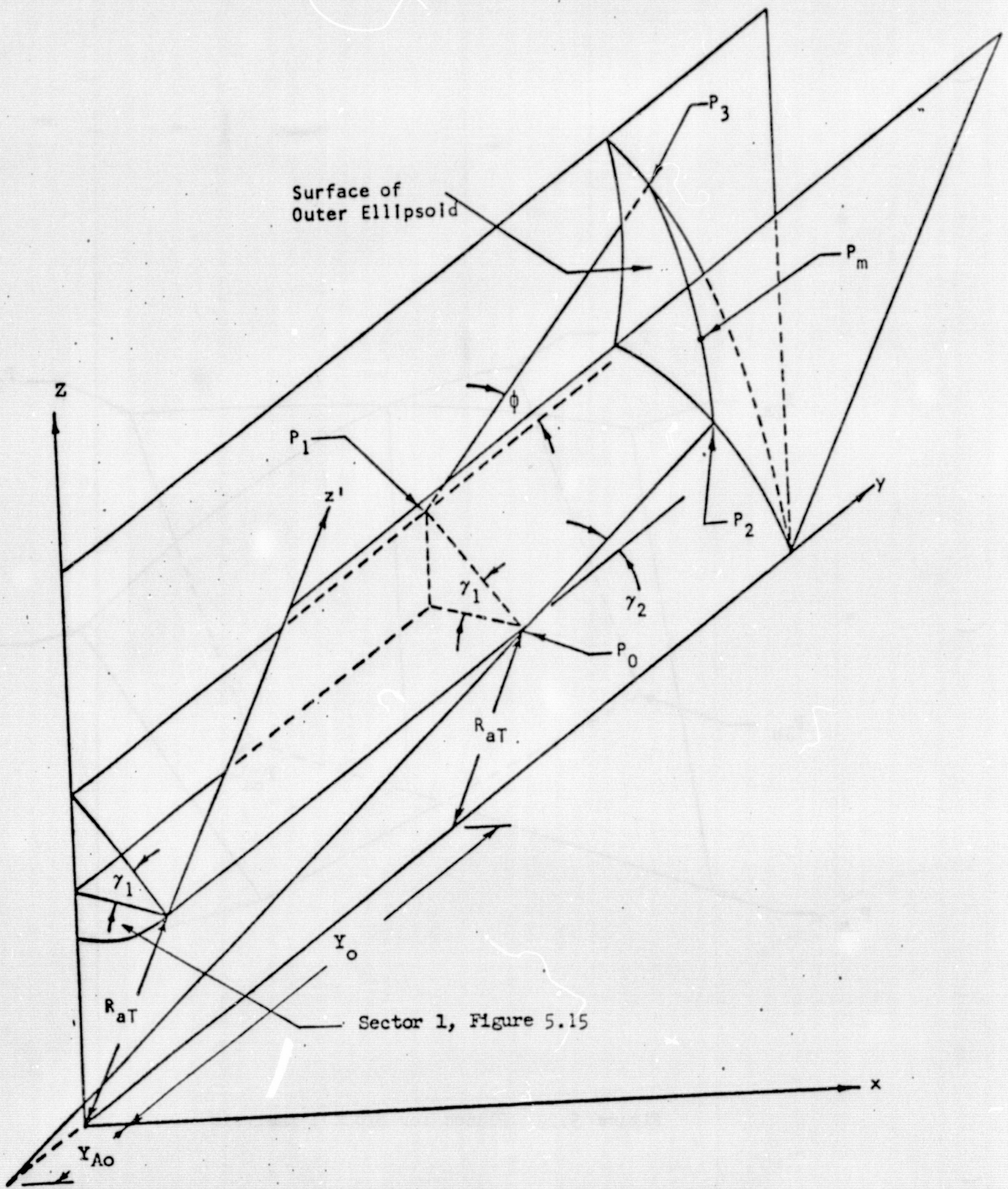


Figure 5.22 P_3 for Subroutine P3SUB

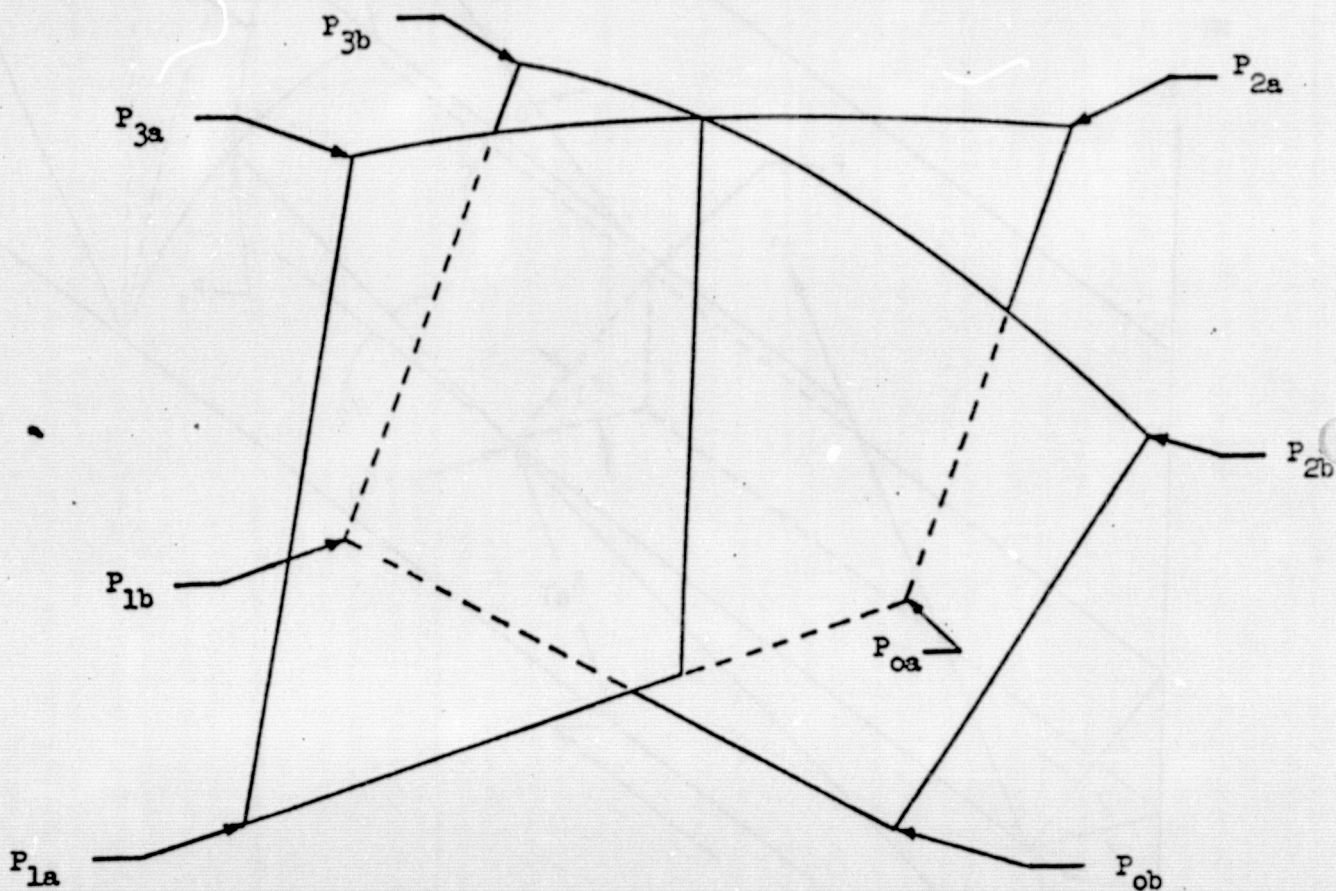
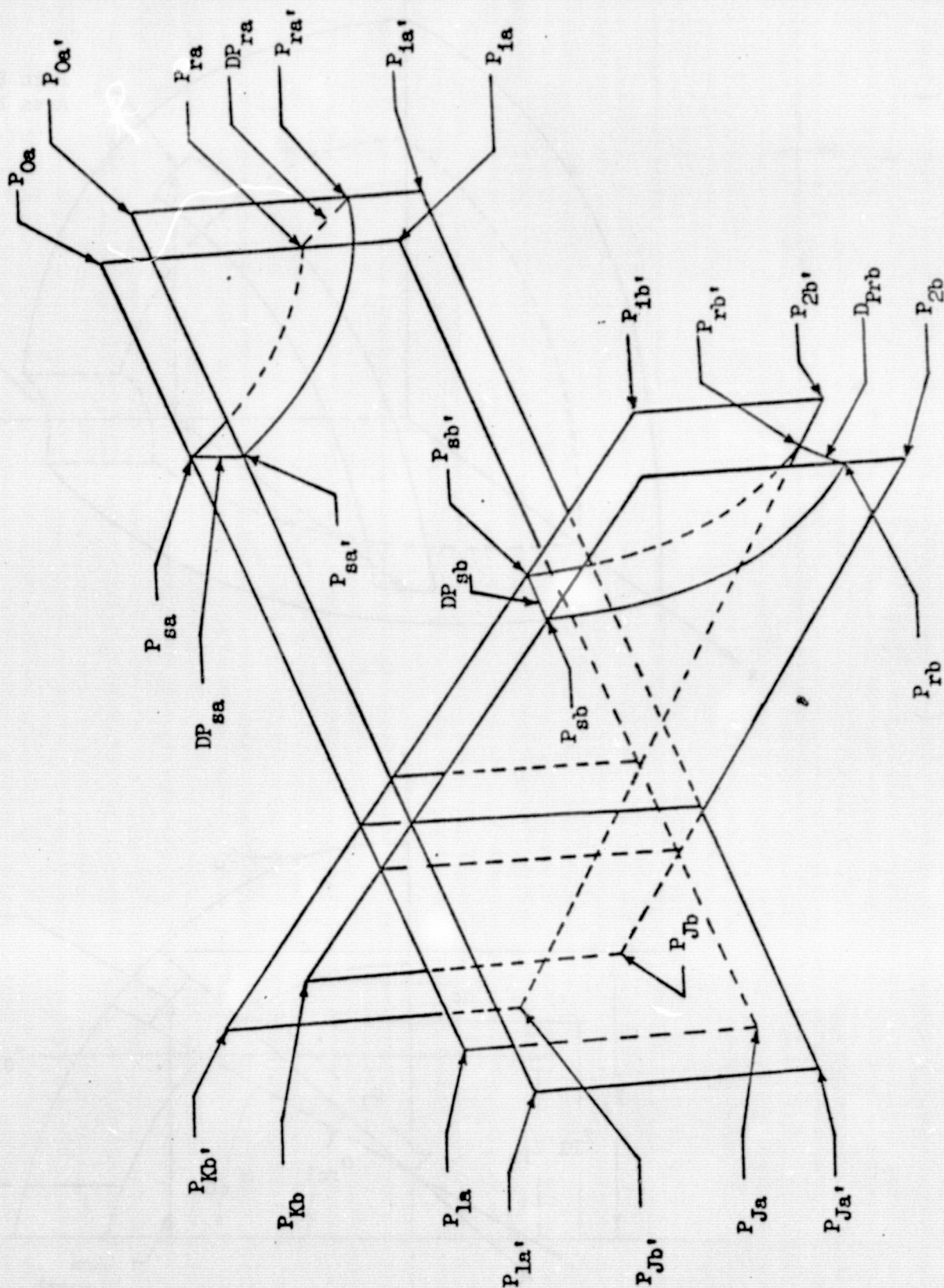


Figure 5.23 Planes for Block 1 Analysis



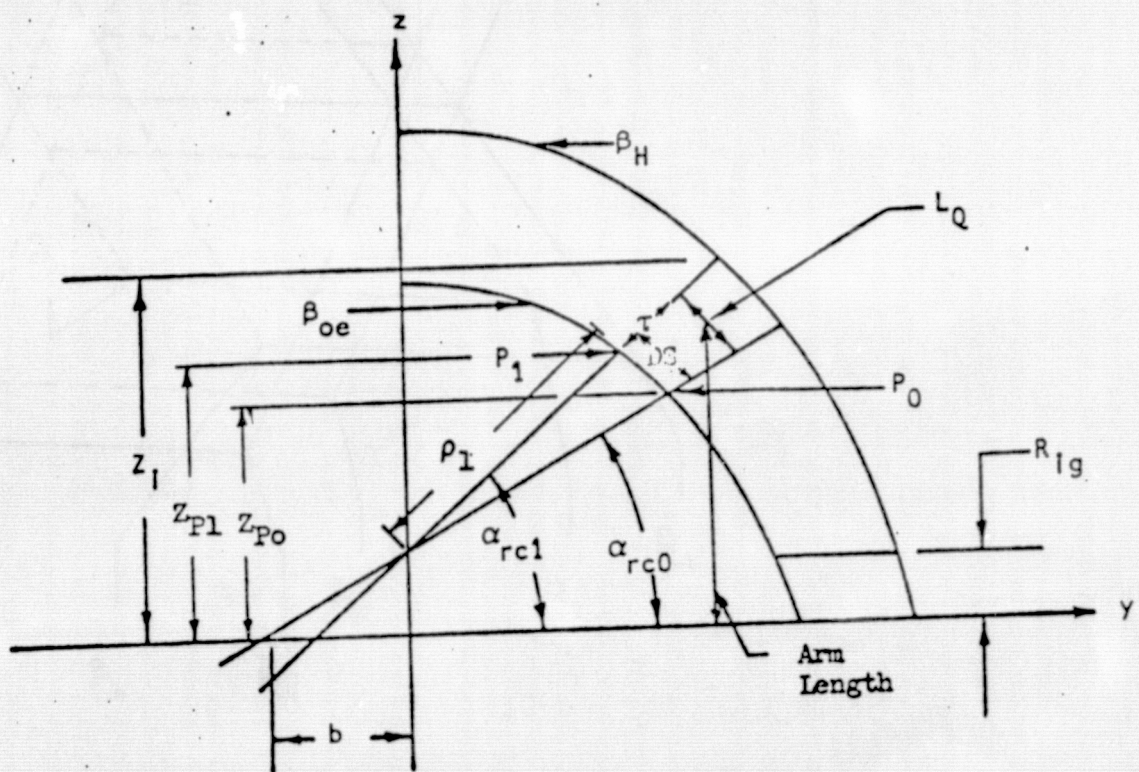
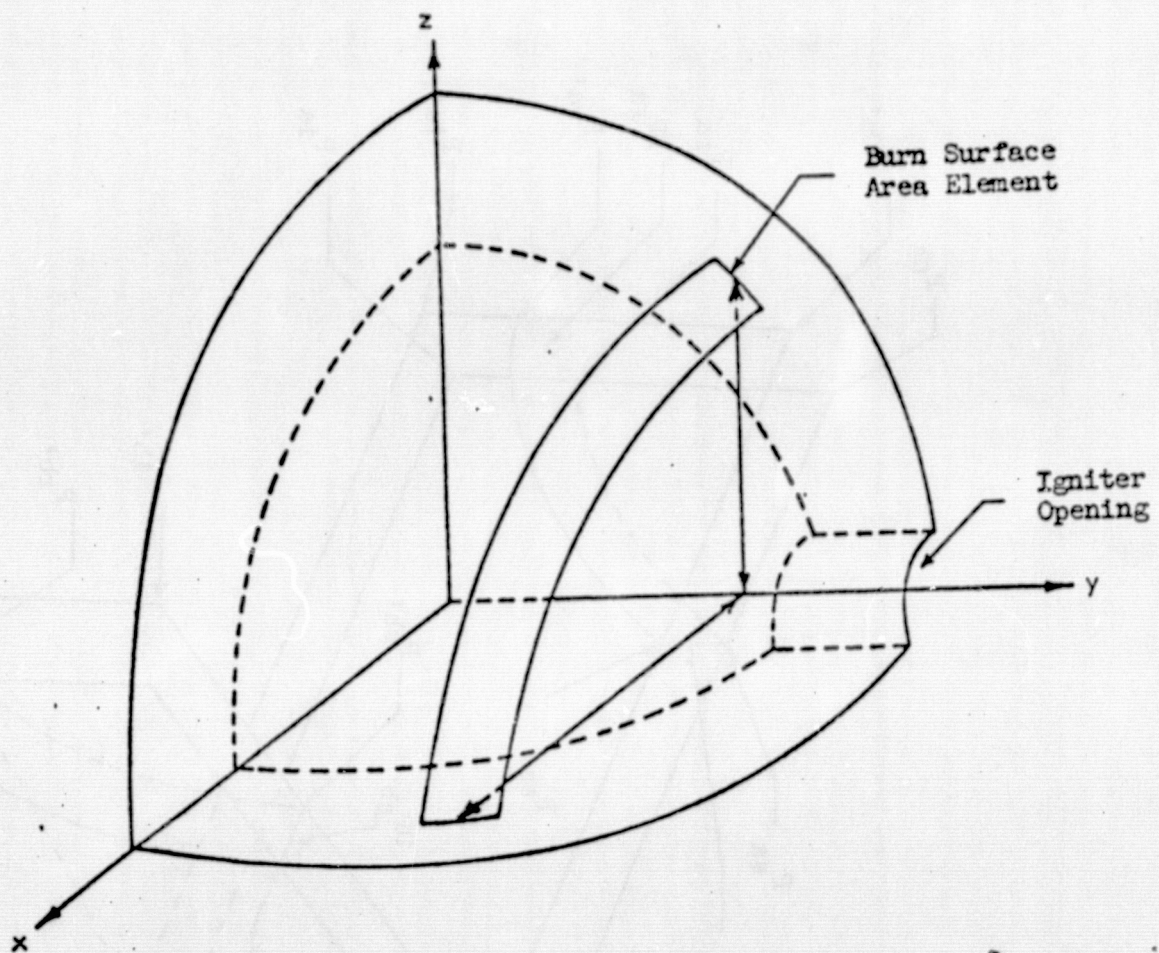


Figure 5.25 Area for Block No. 2A and 2B

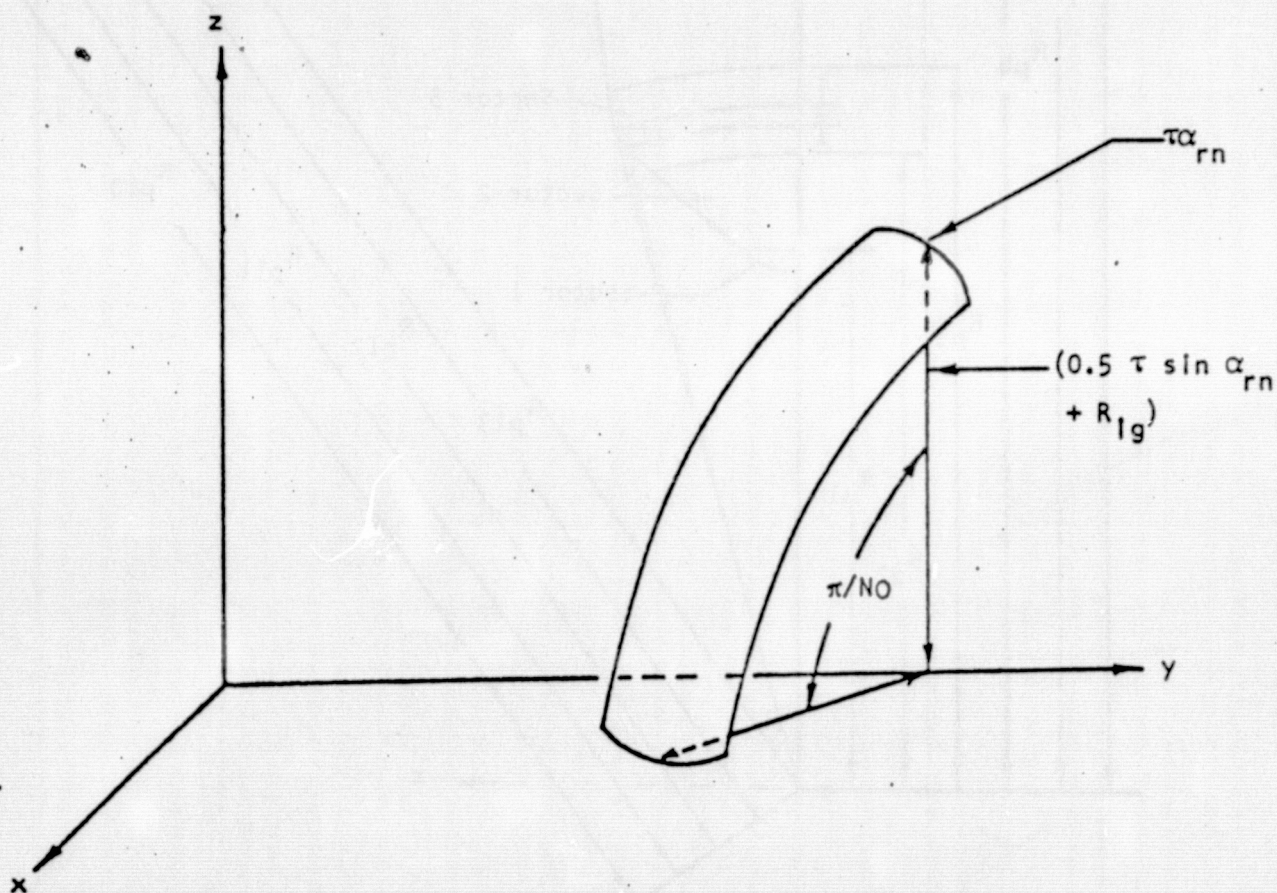
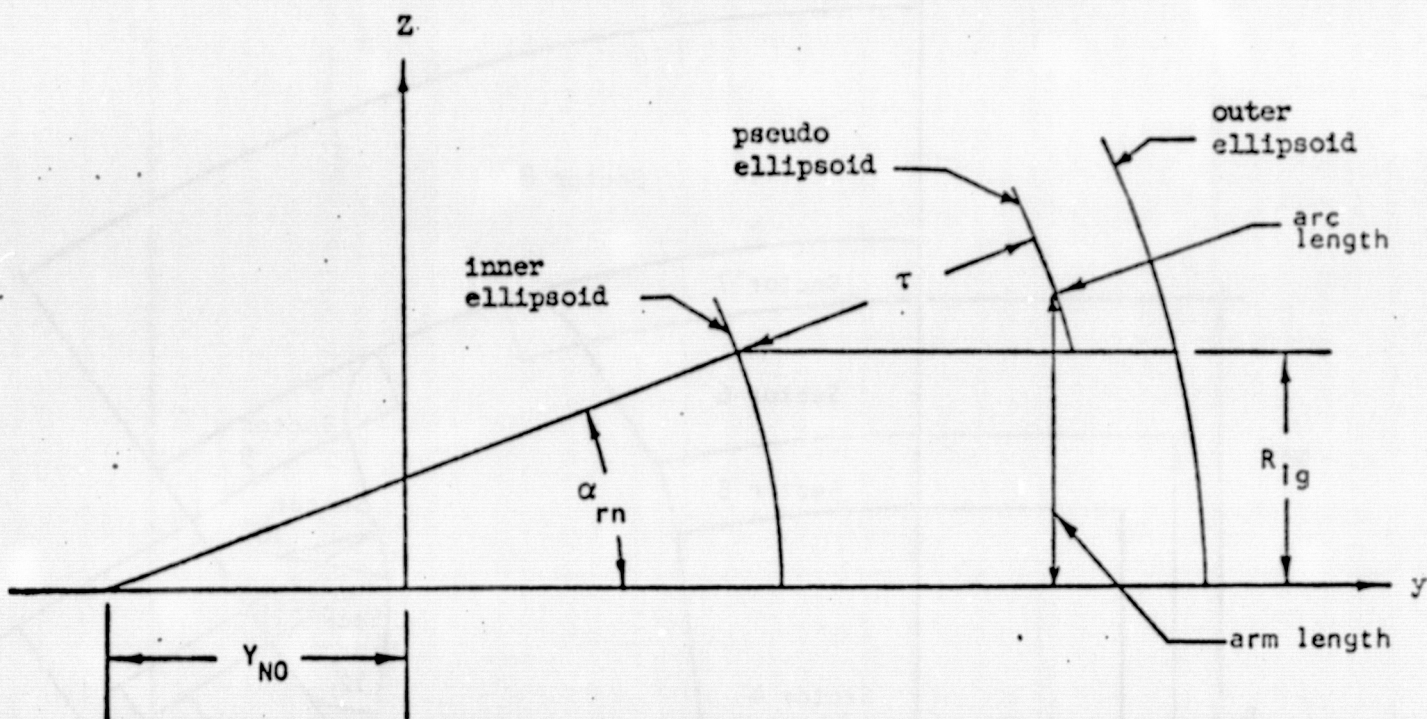


Figure 5.26 A_{lg} for Subroutine AIGSUB

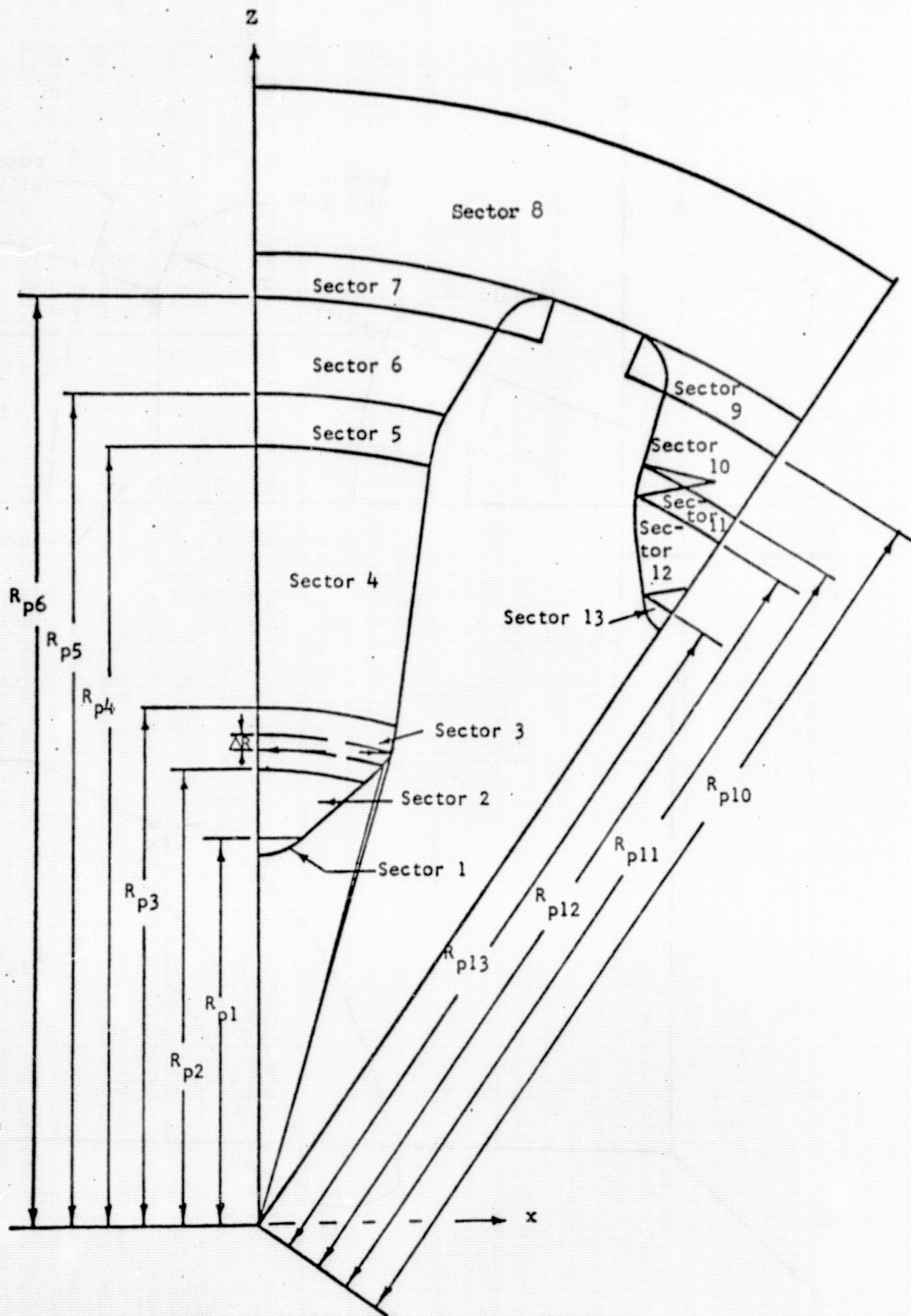


Figure 5.27 Sectors for Block 2B of Fore-Head Section

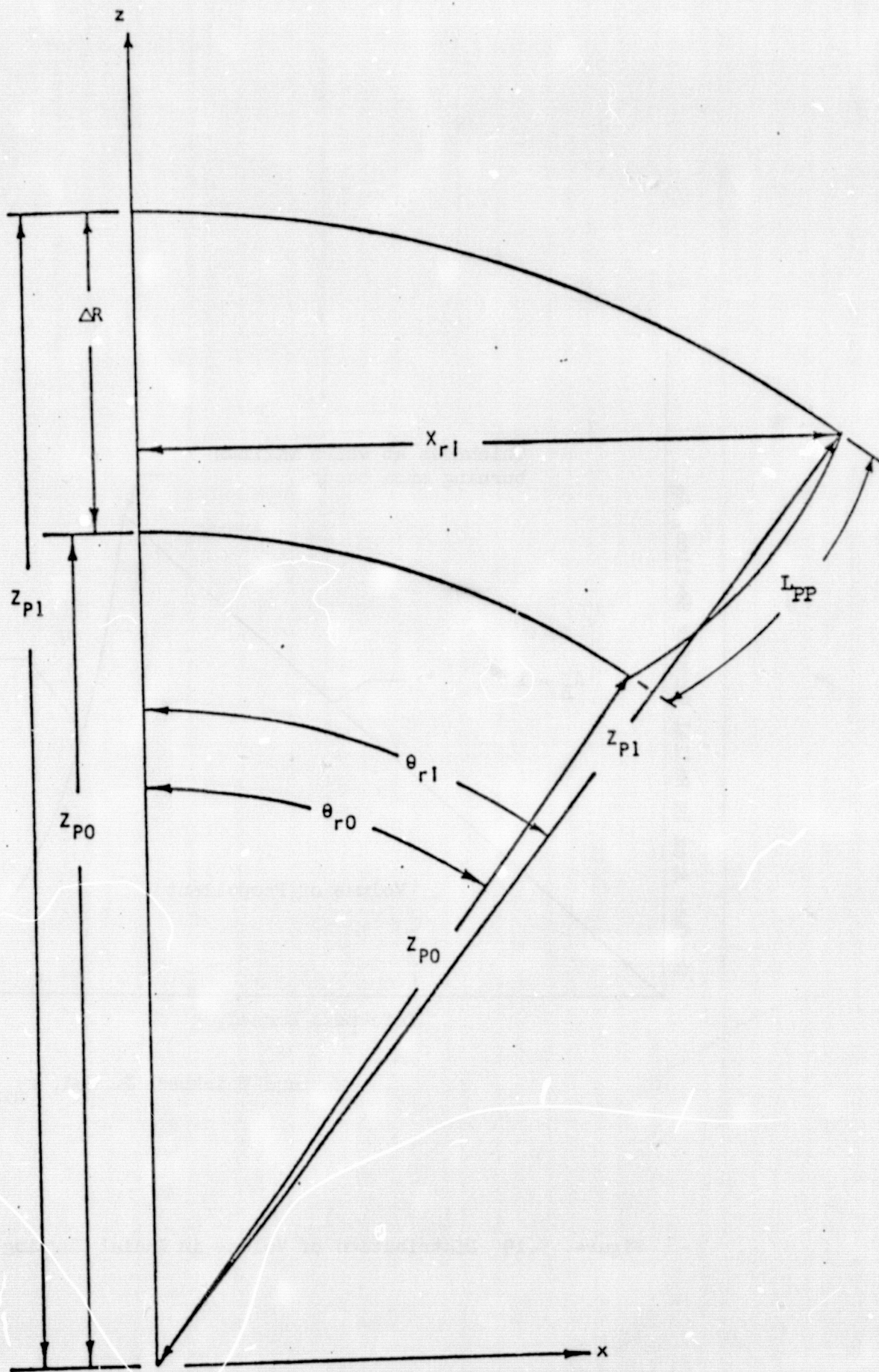


Figure 5.28 Sector for Block 2B

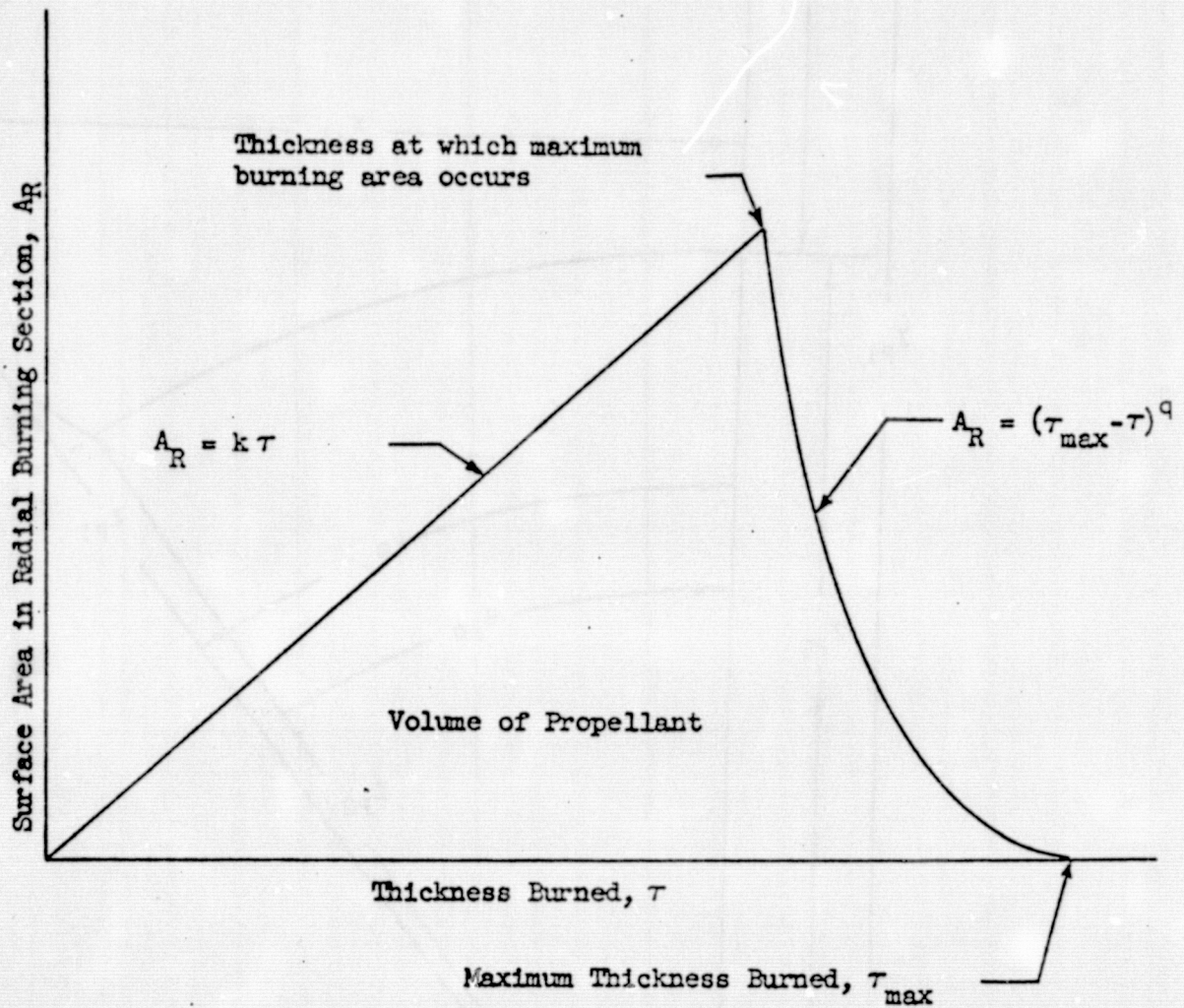


Figure 5.29 Distribution of Volume in Radial Burning Section

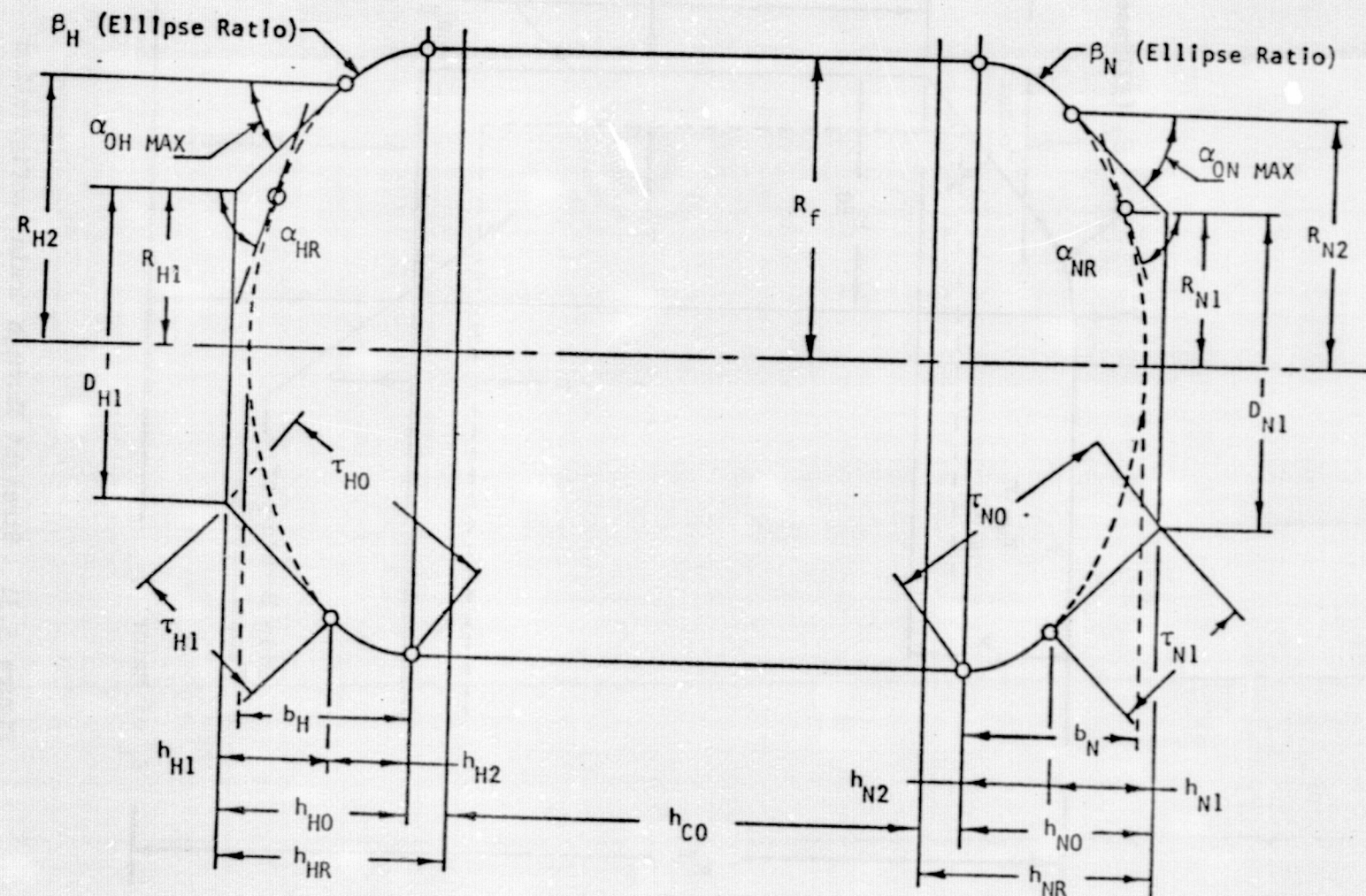


Figure 5.30 Motor Case Longitudinal Constants

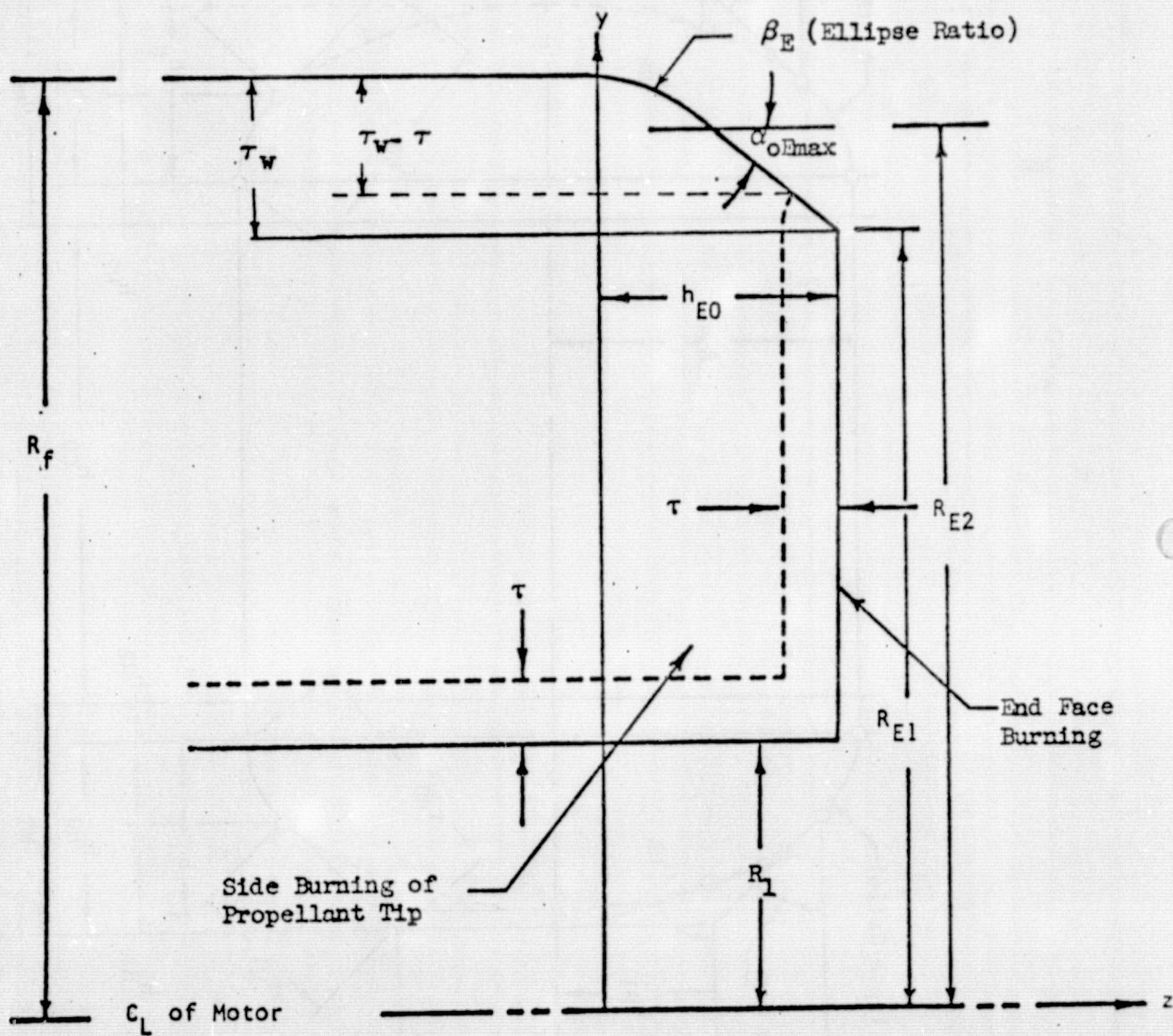


Figure 5.31 Straight Through Grain Configuration

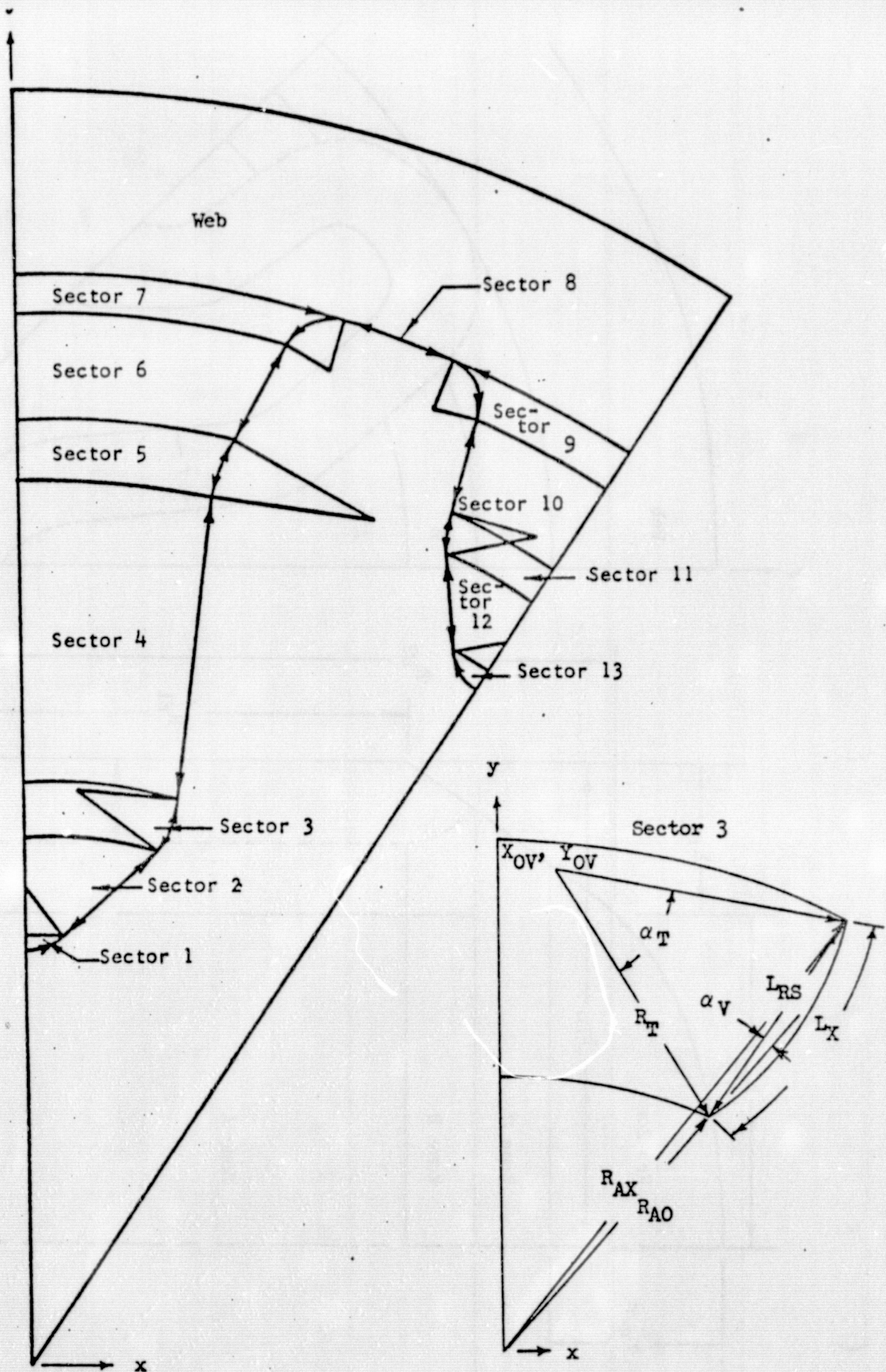
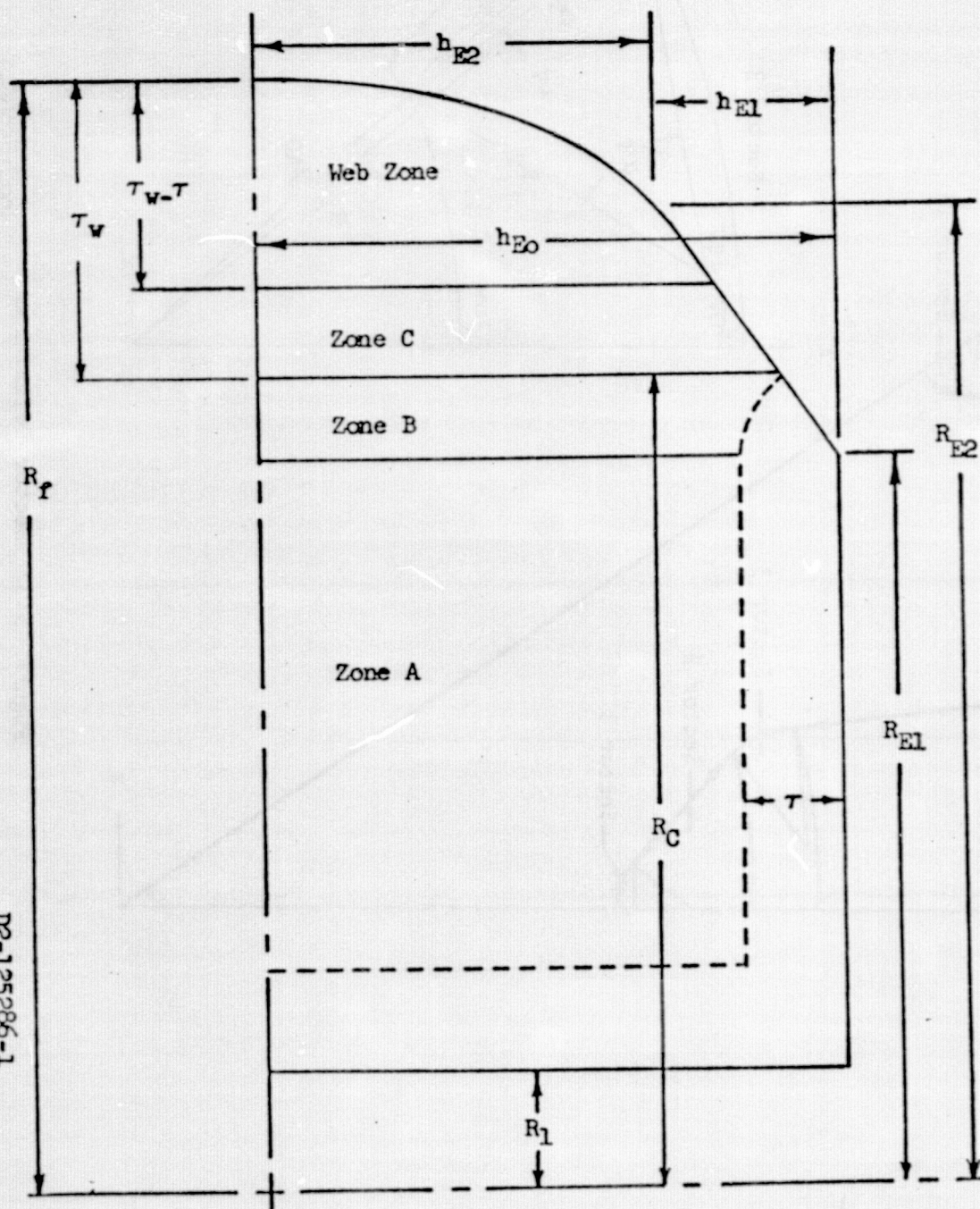
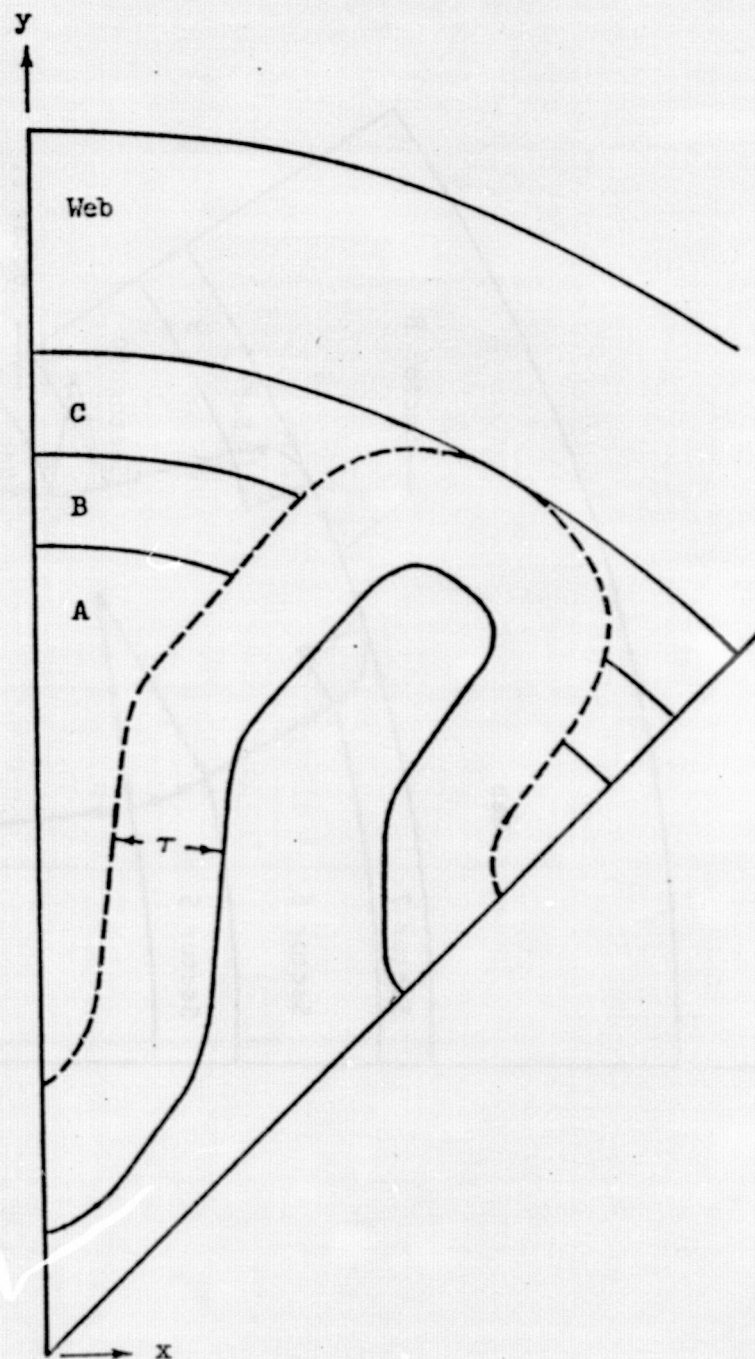


Figure 5.32 Sector Definition

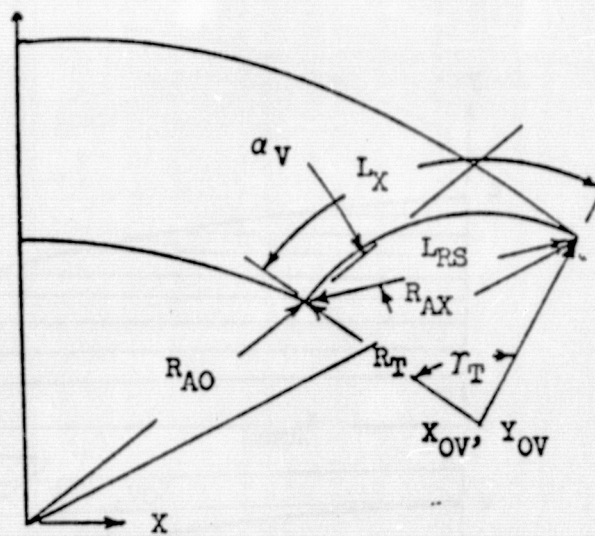
D2-125286-1



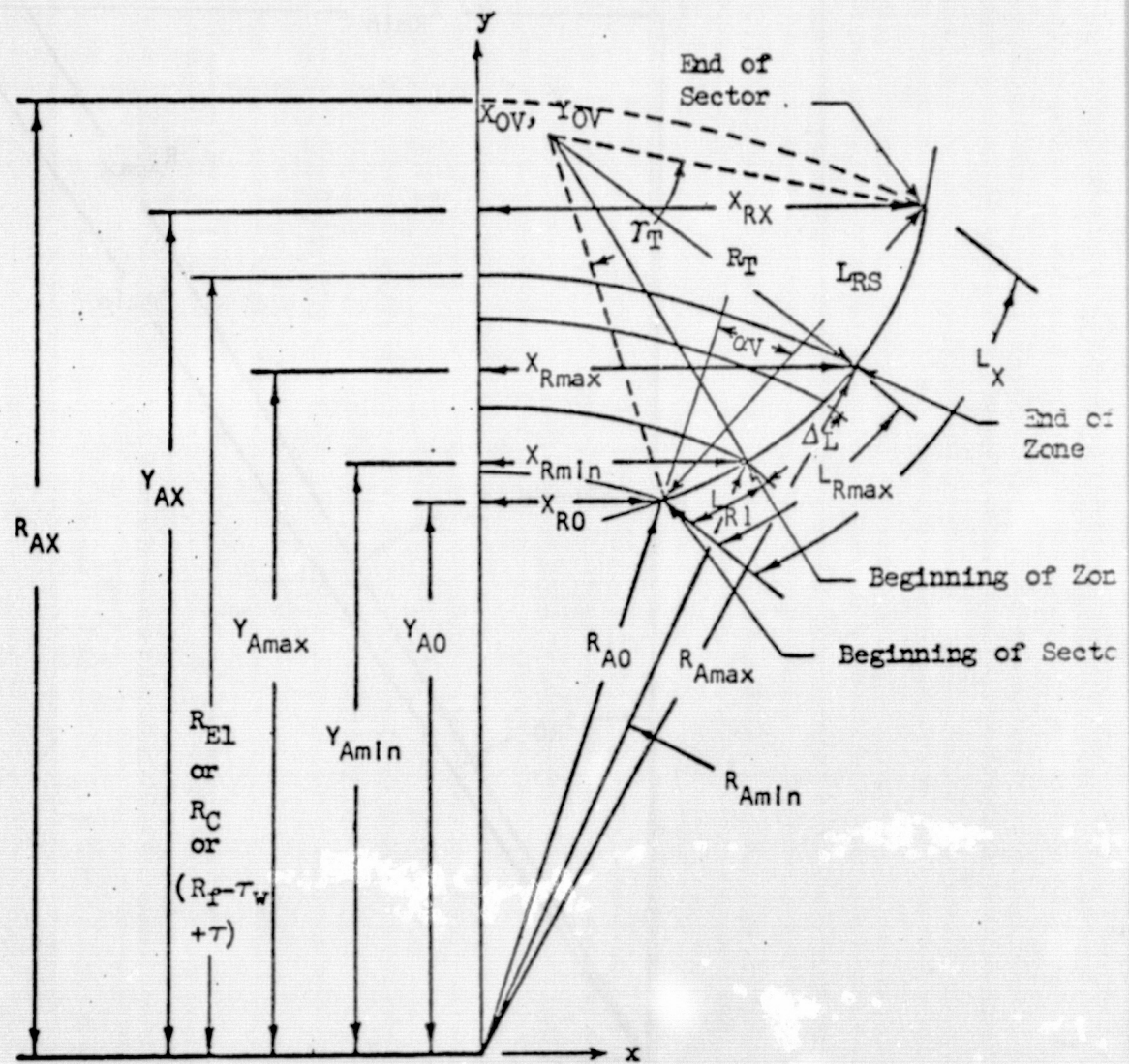
Side View



End View



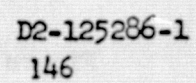
Sectors 5, 7, 9



Sectors 1, 3, 11, 13

Figure 5.34 Sector Parameters, Zones B, C, and Web

D2-125286-1
11c



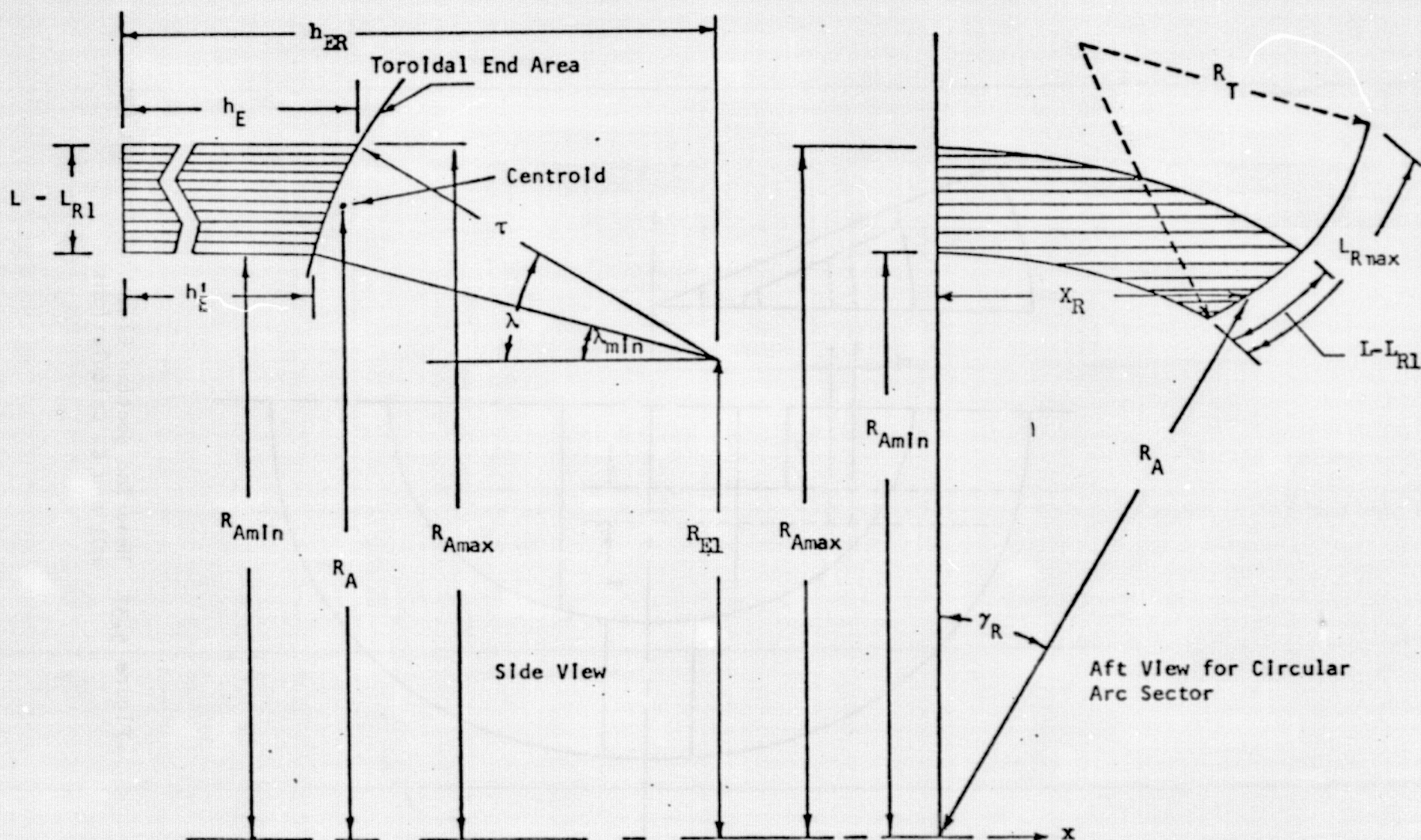


Figure 5.36 Element of Incremental Area for Zone B

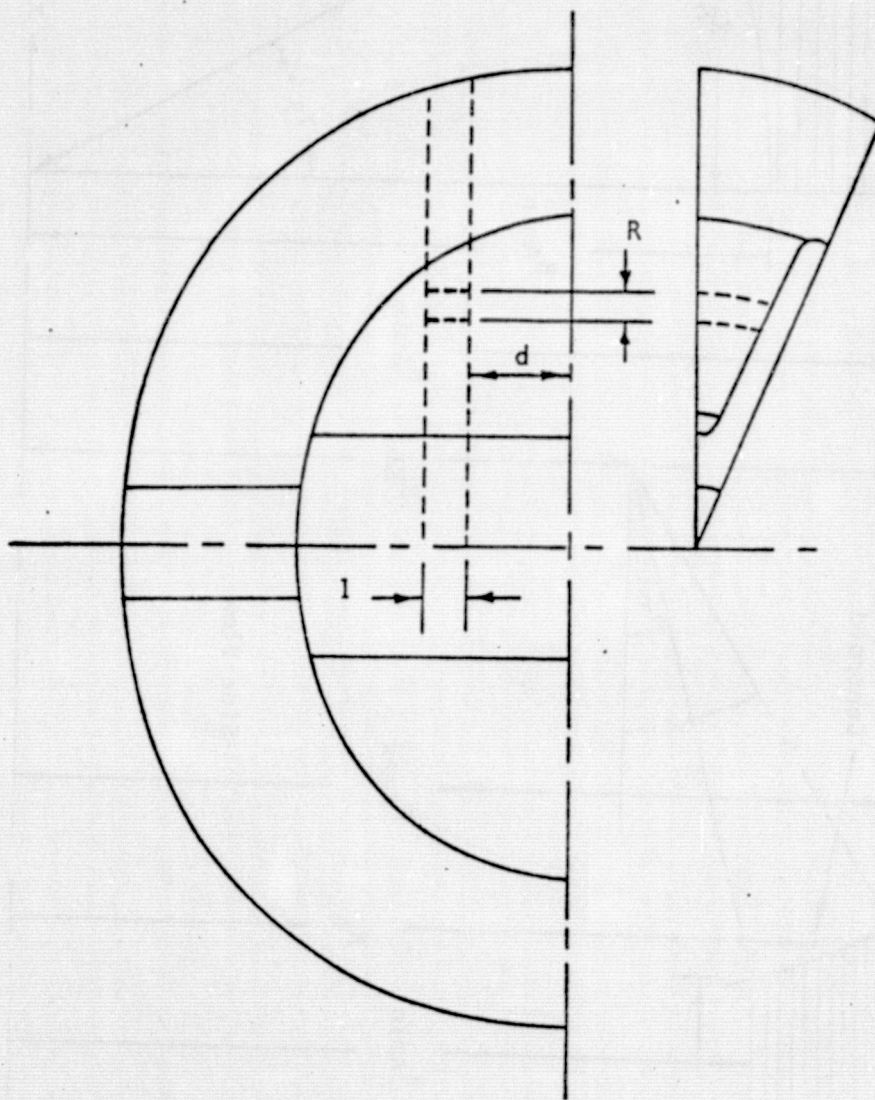


Figure 5.37 Head-End Section Elemental Volumes for MOI and CG Calculations

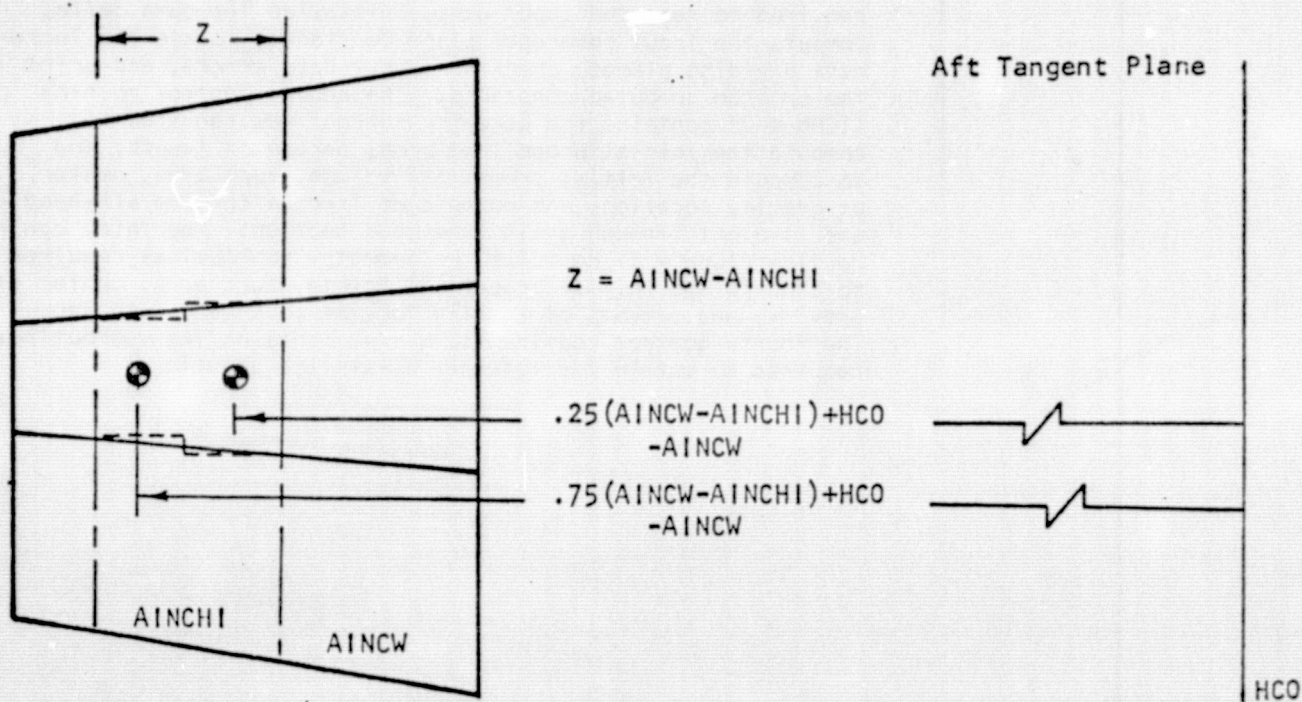
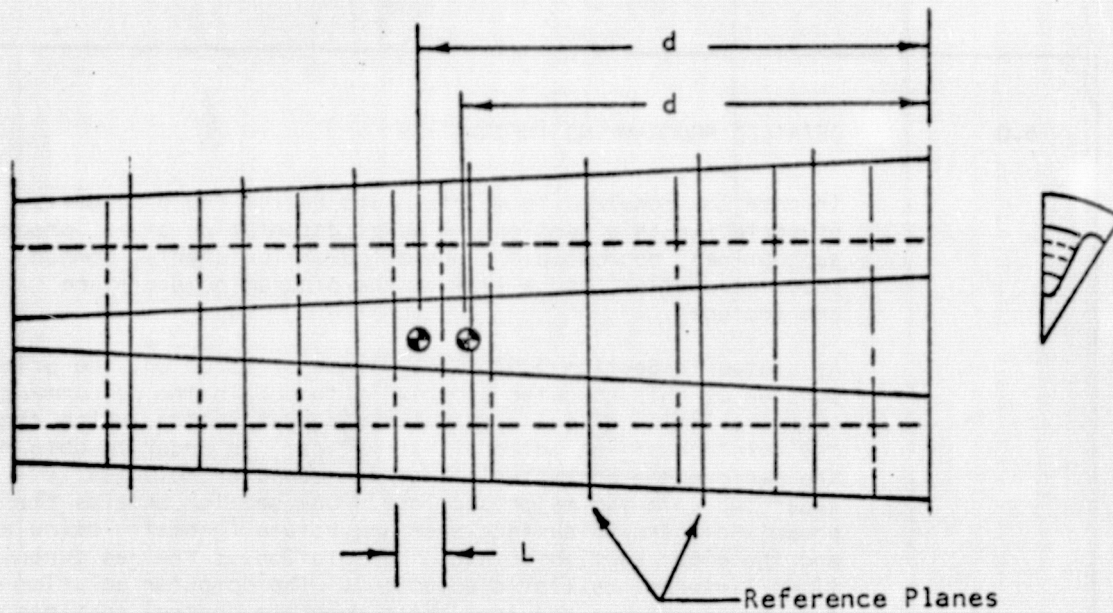


Figure 5.38 Cylindrical Section Elemental Volumes for MOI and CG Calculations

Information required to maintain and revise the program is presented in this section. A brief description of all program subroutines, macroscopic program logic flow charts, common block allocation, and a list of the program nomenclature are included.

As stated in Section 3.0 on the method of solution, the primary purpose of this computer program is to obtain the performance characteristics of solid propellant rocket motors, which requires the solution of the internal ballistics. In order to obtain the performance characteristics, the computer solution is separated into two major sections. One section obtains the propellant burning surface area and volume (geometry calculations), and the other section obtains the solution of the gas dynamic equations (internal ballistic solution). The computer solution of the geometry equations requires three separate control routines and the computer solution of the gas dynamic equations requires a fourth control routine. Each control routine calls the required subroutines and computes values of control parameters. The individual control routines are linked through subroutines LINK1 and LINK2 by a control variable named ICHN. The first control routine is MNCHN1, the second control routine is MNCHN2, the third control routine is MNCHN3, and the fourth control routine is MNCHN4.

The first control routine (ICHN = 1) contains the subroutines required to read the input data, initialize the data cells, compute the input reference plane constants, locate the increment dividing planes, check for input data errors, and print the program computed constants. The second control routine (ICHN = 2) contains the geometry subroutines required to compute the initial propellant area, perimeter length, and to compute the initial propellant volume, burn area, center of gravity location, and moments of inertia for the aft-head and straight through grain fore-head section. The third control routine (ICHN = 3) contains the geometry subroutines required to compute the initial propellant volume, burn area, center of gravity, and moments of inertia for the fore-head with web. The fourth control routine (ICHN = 4) contains the subroutines required to obtain the internal ballistics solution.

- 6.1 Subroutine Description
- This subsection is divided into two parts. Part one (6.1.1) contains a brief description of the purpose of each subroutine and includes cross-references to other sections which give additional, detailed explanations of particularly complex or important routines. Part two (6.1.2) contains a tabular description of the subroutine linkage. The table identifies all lower level subroutines called by a particular routine plus each call that particular subroutine.
- 6.1.1 Subroutine Descriptions
- ACOS - Routine ACOS determines the arc cosine for arguments between 0 and 2π radians.
- AEPSUB - Routine AEPSUB tests for the existence of a sector for the end section. If a sector is burned or does not exist, program control is returned to subroutine ASESUB, otherwise computation proceeds to subroutine AESUB. The sector areas are then added and program control is returned to subroutine SUB (Section 5.2.3.2).
- AESUB - Routine AESUB determines the surface area and initial volume of a sector for the zones in the end sections (Section 5.2.3.2).
- AFPSUB - Routine AFPSUB determines the perimeter length and cross-sectional area of all sectors, except for 8, in the cylindrical section (Section 5.1.1).
- AIBST - Routine AIBST determines the pressure loss and changes in the gas properties between increment dividing planes due to mass addition and area change in the non-steady flow solution of the internal ballistics (Section 4.1.1).
- AIBSUB - Routine AIBSUB determines the pressure loss and changes in the gas properties between increment dividing planes due to mass addition and area change in the solution of the internal ballistics neglecting transient effects (Section 4.1.1).
- AIGSUB - Routine AIGSUB determines the surface area around the igniter opening in the head end section (Section 5.1.2).
- ALRSUB - Routine ALRSUB determines the arc length of a sector in the end section from the minimum point of the sector to a general point along the perimeter of the sector (Section 5.2.3.2).

USE FOR TYPEWRITTEN MATERIAL ONLY

6.1.1 Subroutine Descriptions (Continued)

- ARSSUB - Subroutine ARSSUB determines the chord length between the minimum point of a sector and a general point along the perimeter of the sector (Section 5.2.3.2).
- ASESUB - Subroutine ASESUB sets up the correct equations for subroutines XRSUB and RASUB and assigns for each sector the proper values for the coordinates of the origin of the circular arc (RA0, XA0), the radius of curvature of the sector (RT), and the perimeter length (AL) of the sector (Section 5.2.3.2).
- ASIN - Function ASIN determines the arc sine in radians for the argument angle X.
- ASTSUB - Subroutine ASTSUB sets up the correct equations for subroutine PT1AA to determine the moments of inertia for the block 1 analysis of the head end with web (AJSTP and AJSTB).
- ASUBC - Subroutine ASUBC sets up the correct variables and equations to determine the coordinates (X, Y, and Z) of the points POA, P1A, and P3A for the block 1 analysis in subroutine SC1 (Section 5.2.1.1).
- AWESUB - Subroutine AWESUB determines the total burning surface area of the web zone at thickness TAU and the burning surface area of sector 8 for the end sections (Section 5.2.3.2.4).
- BRAKSB - Subroutine BRAKSB determines the length of the diagonal of the parallelogram that is formed by the intersection of two planes in the block 1 analysis (See Figure 5.23).
- BSUBC - Subroutine BSUBC sets up the correct variables and equations to determine the coordinates (X, Y, and Z) of the points POB, P1B, and P3B on the pseudoellipsoid for the block 1 analysis in subroutine SC1 (Section 5.2.1.1.1).
- CONV - Subroutine CONV determines new iterations value for either AKRST or PCTAB during the start transient calculations for the internal ballistic solution.

USE FOR TYPEWRITTEN MATERIAL ONLY

6.1.1 Subroutine Descriptions (Continued)

DPRASB - Subroutine DPRASB determines the distance between the points P_{ra} and P_{ra}' ; P_{sa} and P_{sa}' ; P_{rb} and P_{rb}' ; and P_{sb} and P_{sb}' that lie on the planes produced in sectors 3A and 3B or 11A and 11B in the block 1 analysis of subroutine SCI (Section 5.2.1.1.2).

ENDCSB - Subroutine ENDCSB determines the coefficients of a fourth degree polynomial equation obtained by the intersection of an ellipse and a circle, CAE, CBE, CCCE, CCVE, CDCE, CDVE, CECE, and CEVE and determines the constants RE1, RE2, ALFE, ALFEM, HE1, HE2, and HEO that define the geometry of the fore-head and aft-head sections (Section 5.2.3.1).

FDGRE - Subroutine FDGRE solves for the roots of a fourth degree polynomial equation by reducing to the form

$$X^4 + A X^2 + B X + C = 0$$

by the substitution

$$Y = (X - P/R)$$

and then solving the resultant cubic

$$T^3 + \frac{A}{2} T^2 + \frac{(A^2 - 4C)}{16} T - \frac{B}{64} = 0$$

GAMA2S - Subroutine GAMA2S determines the angle between the Y-axis and a line normal to the ellipse

$$\left(\frac{Y}{BOE}\right)^2 + \left(\frac{Z'}{AOE}\right)^2 = 1$$

(which is defined by the ellipse ratio β_{OE}) at the point $Z' = RAT$ (Section 5.2.1.1.1).

GAMSUB - Subroutine GAMSUB determines the angle γ_1 between a line normal to the perimeter in the X-Z plane, and a line normal to the line segment RAT which is a radial vector from the motor axis to a point on a sector perimeter (Section 5.2.1.1.1).

USE FOR TYPEWRITTEN MATERIAL ONLY

6.1.1 Subroutine Descriptions (Continued)

- HAPSBC - Subroutine HAPSBC sets up the correct equations to determine the coordinates (XOP, YOP, ZOP) of the points POA', P1A', and P3A' on the pseudoellipsoid for block 1 analysis of the intersecting planes for sectors 3 and 11 in subroutine SCI (Section 5.2.1.1.2).
- HASUBC - Subroutine HASUBC sets up the correct variables to determine the coordinates (XO, YO, ZO) of the points POA, P1A, and P3A that lie on the pseudoellipsoid produced in the block 1 analysis of the intersecting planes for sectors 3 and 11 in subroutine SCI (Section 5.2.1.1.2).
- HBPSBC - Subroutine HBPSBC sets up the correct equations and variables to determine the coordinates (XOP, YOP, ZOP) of the points POB', P1B', and P3B' that lie on the pseudoellipsoid produced in the block 1 analysis of the intersecting planes for sectors 3 and 11 in subroutine SCI (Section 5.2.1.1.2).
- HDNSUB - Subroutine HDNSUB sets up the correct variables to perform the block 1, 2A, 2B, and 3 analysis of the head-on with web (Section 5.2.1).
- HESUB - Subroutine HESUB determines the length of the zones in the end sections from which the incremental surface areas in subroutine AESUB are determined (Section 5.3.2).
- IBM - Subroutine IBM initializes many of the program constants, variables, and indicators for the first pass through the program. IBM also calls LINK1 and LINK2 which access the four main control routines.
- IBMOUT - Subroutine IBMOUT controls the program output after completion of each ballistic solution iteration.
- INPT - Subroutine INPT controls the internal ballistics data input through Namelist IBDATA.
- LBSUB - Subroutine LBSUB determines the length (b) of the Y intercept produced by the intersection of the line Y and the line normal to the ellipse at P0 (Section 5.2.1.2).

6.1.1 Subroutine Descriptions (Continued)

- LINK1 - Subroutine LINK1 controls access to the three control routines, MNCHN1, MNCHN2, and MNCHN3, which generate the program geometry data.
- LINK2 - Subroutine LINK2 controls access to subroutine MNCHN4 which obtains the internal ballistics solution.
- LOOKUP - Subroutine LOOKUP performs the table lookups for the reference plane perimeter and radius of gyration and the end section burn areas, centers of gravity, and moments of inertia at the beginning of each time point solution in subroutine MNCHN4.
- LPDAPS - Subroutine LPDAPS sets up the correct variables to determine the perimeter length, ALP, and the cross-sectional fuel area, AFP, of the propellant tips in subroutine AFPSUB for the cylindrical section reference planes (Section 5.1.1).
- LPT0 - Subroutine LPT0 determines the perimeter length AL7 and area of the anisotropic propellant in sectors 7 and during the motor tail-off interval (Section 4.1.1).
- MNCHN1 - Subroutine MNCHN1 is the control routine which computes the constants that define the geometry of the grain cross-section and longitudinal configuration for use by the second and third core loads, checks for data errors, and prints geometry constants.
- MNCHN2 - Subroutine MNCHN2 is the control routine which determines the initial fuel area (AFF) and port area (AP) for the cylindrical section reference planes and determines the burn area and initial fuel volume of the straight through grain fore-head and aft-head sections as a function of distance burned.
- MNCHN3 - Subroutine MNCHN3 is the control routine which determines the initial fuel volume and burn area as a function of time for the head end with web.
- MNCHN4 - Subroutine MNCHN4 is the control routine which obtains the internal ballistic solution (Section 5.2.2).

USE FOR TYPEWRITTEN MATERIAL ONLY

6.1.1 Subroutine Descriptions (Continued)

- MODTSB - Subroutine MODTSB modifies the value of TIME and the value of the nozzle throat diameter after convergence of the ballistic solution iteration and initializes X and Y reference planes for interpolation of the Increment dividing planes in subroutine TISUB.
- MSISUB - Subroutine MSISUB determines the location of the center of gravity and the polar and rectangular moment of inertia for the block 2A analysis of the head-end with web.
- MTISUB - Subroutine MTISUB determines the location of the center of gravity and the polar and rectangular moment of inertia for the block 2B analysis of the head end with web.
- PLNCNS - Subroutine PLNCNS calculates the geometry constants of each cylindrical section reference plane (Section 5.1.1).
- PLNLCS - Subroutine PLNLCS checks for reference plane input data errors, prints appropriate diagnostic comments, and flags the program for case termination if an error exists.
- POSUB - Subroutine POSUB determines the coordinates (X0, Y0, Z0) of the point P0 that is located on the intersection of the inner ellipsoid with the core (Section 5.2.1.1.1).
- PT1AA - Subroutine PT1AA determines the polar and rectangular moment of inertia for the cylindrical section, straight through grain end sections, and the block 1 analysis of the head end with web (Section 5.3).
- P1SUB - Subroutine P1SUB determines the coordinates (X1, Y1, and Z1) of the point P1 that is located on the Z axis along a line through point P0 and normal to the sector perimeter (Section 5.2.1.1.1).
- P3SUB - Subroutine P3SUB determines the coordinates (X3, Y3, and Z3) of the point P3 that is located in the Y-Z plane and on the outer ellipse (Section 5.2.1.1.1).
- RASUB - Subroutine RASUB determines the length of a radius vector from the motor axis to a point on the perimeter for each sector in the end sections with a straight through web (Section 5.2.3.2).

USE FOR TYPEWRITTEN MATERIAL ONLY

6.1.1 Subroutine Descriptions (Continued)

- RASUBB - Subroutine RASUBB determines the length of a radius vector from the motor axis to a general point in a sector for the block 1 analysis of the head end with web (Section 5.2.1.1.1).
- RBSTSB - Subroutine RBSTSB determines the initial estimate of the burn rate coefficient, AKRST, for each time increment during the start transient interval and performs the table look-ups for PH and AKRST (Section 4.3.3).
- RBSUB - Subroutine RBSUB determines the propellant burning rate at each increment dividing plane (Section 4.3.2).
- RBVSUB - Subroutine RBVSUB checks the validity of the burning rate equation constants and prints appropriate diagnostic comments.
- RCSUB - Subroutine RCSUB determines the value of the radial distance from the motor axis to the intersection of the aft-end burning surface and the motor case for any configuration (Section 5.2.3.2).
- RGISUB - Subroutine RGISUB determines the radius of gyration for the incremental thin shells of the head end with web.
- ROE1SB - Subroutine ROE1SB determines the radius of curvature, ρ_1 , at the point P_1 on the pseudoellipsoid for the block 2A analysis in subroutine SCI (Section 5.2.1.2).
- ROPSB - Subroutine ROPSB sums the values of the Y coordinates P_0 (Y_{0A} and Y_{0B}) or P_0' (Y_{0A}' and Y_{0B}') for the A and B planes from which the block 1 surface area is obtained in subroutine SCI (Section 5.2.1.1.1).
- SCI - Subroutine SCI is the control routine to determine the surface area and initial volume of the propellant tip for the block 1 analysis of the head end with web (Section 5.2.1.1).
- SCTOR1 - Subroutine SCTOR1 is the control routine to determine the surface of the pseudoellipsoid for the block 2A analysis of the head end with web (Section 5.2.1.2).

USE FOR TYPEWRITTEN MATERIAL ONLY

6.1.1 Subroutine Descriptions (Continued)

- SCTOR2 - Subroutine SCTOR2 is the control routine to determine the surface area on the pseudoellipsoid of the projected propellant core in the block 2B analysis of the head-end with web (Section 5.2.1.3).
- SD1D13 - Subroutine SD1D13 determines the center of gravity and moments of inertia of the straight through grain end sections (Section 5.3).
- SEGSUB - Subroutine SEGSUB is the cylindrical section control routine to determine the mass generation rate, port area, perimeter length, and cross-sectional fuel area for each increment dividing plane and mass addition region during the internal ballistic solution (Section 5.2.2).
- SETPH - Subroutine SETPH is the internal ballistic solution control routine to obtain convergence on the fore head pressure by matching the grain discharge flow with the nozzle flow determined from the nozzle pressure. The performance calculations for thrust, total impulse, thrust coefficient, etc., are included in subroutine SETPH (Section 4.2).
- SLOT - Subroutine SLOT determines the gas properties at the discharge section of a slot between grain segments for the non-steady flow solution of the internal ballistics (Section 4.1.2.2).
- SQRT - Subroutine SQRT is a modification of the IBSYS-13 monitor library routine to determine the square root of negative numbers. Only the positive value of the argument is transferred to the designated register.
- S2SK - Subroutine S2SK determines the sector surface area on the pseudoellipsoid of the projected grain cross-section for the block 2B analysis in subroutine SCTOR2 of the head end with web (Section 5.2.1.3).
- TDGRE - Subroutine TDGRE determines the largest real root of a third degree polynomial for the argument X.

USE FOR TYPEWRITTEN MATERIAL ONLY

THE **BOEING** COMPANY

6.1.1 Subroutine Descriptions (Continued)

- THETAR - Subroutine THETAR determines the angle between the Z-axis and the line segment PAT for the block 1 analysis in subroutine DCI of the head end with web (Section 5.2.1.1.1).
- TISUB - Subroutine TISUB determines the value of DELT for the steady state internal ballistic solution neglecting transient effects and modifies the values of thickness burned in each increment dividing plane after the ballistic solution is converged for each time point.
- TRAN - Subroutine TRAN transfers the geometry constants from the permanent common storage location to the working array common storage location.
- VFPPSB - Subroutine VFPPSB determines the port volume of each cylindrical section segment and sums the segment port volumes to obtain the total cylindrical section port volume.
- VOLSUB - Subroutine VOLSUB is the control routine which determines the initial volume for the block 3 analysis of the head-end with web (Section 5.2.1.4).
- VSEC - Subroutine VSEC determines the volume produced by the difference of volumes of two oblate spheroids, minus the volume of the igniter hole in the block 3 analysis of the head end with web (Section 5.2.1.4).
- VSTRSB - Subroutine VSTRSB determines the initial core volume that is present in the head end with web for the block 3 analysis (Section 5.2.1.4).
- XLIN - Subroutine XLIN is the internal ballistics program linear interpolation routine.
- XRSUB - Subroutine XRSUB determines the X-coordinate of a general point on the perimeter of a sector in the end sections (Section 5.2.3.2).
- XRSUBB - Subroutine XRSUBB determines the X-coordinate of the RAT line segment which is a radial vector from the motor axis to a point on a sector perimeter for the analysis of the head end with web (Section 5.2.1.1.1).

USE FOR UNWRITTEN MATERIAL ONLY

6.1.1 Subroutine Descriptions (Continued)

- XRTHR - Subroutine XRTHR is a set up subroutine that uses subroutine XRSUBB to obtain the X-coordinate of a point located on the perimeter of a sector in the block 2B analysis of the head end with web. The angle θ_r between the Z axis and a line from the motor axis to a general point in a sector is also determined (Section 5.2.1.3).
- YPSUB - Subroutine YPSUB determines the Y-coordinate of the points P0 and P3 which are located on the surface of the inner and outer ellipsoids respectively for the block 2 analysis of the head end with web (Section 5.2.1.2).
- ZERODV - Function ZERODV establishes a zero value for the quotient of a division term which has a zero denominator.
- ZISUB - Subroutine ZISUB determines the Z-coordinate produced by the intersection of the outer ellipse and the line normal to the ellipse at P1 (Section 5.2.1.2).

USE FOR TYPEWRITTEN MATERIAL ONLY

USE FOR TYPEWRITTEN MATERIAL ONLY

6.1.2

Subroutine Linkage Table

<u>NAME</u>	<u>LOWER-LEVEL CALLS</u>	<u>CALLED BY</u>
ACØS	SQRT	AESUB AFPSUB AIGSUB ALRSUB AWESUB ENDCSB FDGRE
		GAMA2S LPTØ PLNCNS PISUB P3SUB RASUB
		RASUBB SDID13 THETAR XRTHR YPSUB ASIN NSCE
AEPSUB	AESUB	ASESUB
AESUB	ACØS ALRSUB ARSSUB HESUB RASUB SQRT XRSUB	AEPSUB
AFPSUB	ACØS SQRT ASIN ZERØDV	LPDAPS
AIBST	AIBSUB SQRT XLIN ZERØDV	AIBSUB MNCHN4
AIBSUB	AIBST SQRT	AIBST MNCHN4 SEGSUB
AIGSUB	ACØS SQRT	SCTØR1
ALRSUB	ACØS SQRT	AESUB
ARSSUB	SQRT	AESUB
ASESUB	AEPSUB AWESUB RCSUB	ENDCSB MNCHN2
ASIN	ACØS	PLNCNS AFPSUB

6.1.2

Subroutine Linkage Table (Continued)

<u>NAME</u>	<u>LOWER LEVEL CALLS</u>	<u>CALLED BY</u>
ASTSUB	PTIAA ENDCSB	MNCHN3
ASUBC	GAMA2S GAMSUB P0SUB P1SUB P3SUB RASUBB	THETAR TRAN XRSUBB SCI
AWESUB	AC0S HESUB SQRT	ASESUB
BRASB	SQRT	SCI
BSUBC	GAMA2S GAMSUB P0SUB P1SUB P3SUB RASUBB	THETAR TRAN XRSUBB SCI
C0NV		SETPH
DPRASB	SQRT	SCI
ENDCSB	AC0S ASESUB LPDAPS SQRT	MNCHN1 ASTSUB

USE FOR TYPEWRITTEN MATERIAL ONLY

6.1.2

Subroutine Linkage Table (Continued)

<u>NAME</u>	<u>LOWER-LEVEL CALLS</u>	<u>CALLED BY</u>
FDGRE	ACØS SQRT	RCSUB
GAMA2S	ACØS SQRT	ASUBC BSUBC HAPSBC HASUBC HBPSBC HBSUBC
GAMSUB		ASUBC BSUBC HAPSBC HASUBC HBPSBC HBSUBC
HAPSBC	GAMA2S GAMSUB PØSUB P1SUB P3SUB RASUBB	THETAR TRAN XRSUBB SCI
HASUBC	GAMA2S GAMSUB PØSUB P1SUB P3SUB RASUBB	THETAR TRAN XRSUBB SCI
HBPSBC	GAMA2S GAMSUB PØSUB P1SUB P3SUB RASUBB	THETAR TRAN XRSUBB SCI
HBSUBC	GAMA2S GAMSUB PØSUB P1SUB P3SUB RASUBB	THETAR TRAN XRSUBB SCI

USE FOR TYPEWRITTEN MATERIAL ONLY

6.1.2

Subroutine Linkage Table (Continued)

<u>NAME</u>	<u>LOWER-LEVEL CALLS</u>	<u>CALLED BY</u>
HDNSUB	SCI SCTØR1 SCTØR2 VØLSUB	MNCHN3
HESUB	SQRT	AESUB AWESUB PT1AA SD1D13
IBM	LINK1 LINK2	
IBMØUT	FAMCAL ØUTPUT MØDTSB	MNCHN4
INPT	DATLØC SQRT	LINK1
LBSUB		SCTØR1 S2SK
LINK1	INPUT MNCHN1 PCHWRT MNCHN2 INTREC MNCHN3	IBM
LINK2	NSCE MNCHN4	IBM
LØØKUP		MNCHN4
LPDAPS	AFPSUB PT1AA	ENDCSB MNCHN2
LPTØ	ACØS SQRT TRAN	SEGSUB
MACH		NSCE SUBSØN
MNCHN1	ENDCSB PLNCNS PLNLCS	RBVSUB LINK1

6.1.2

Subroutine Linkage Table (Continued)

<u>NAME</u>	<u>LOWER-LEVEL CALLS</u>	<u>CALLED BY</u>
MNCHN2	ASESUB LPDAPS PTIAA RCSUB TRAN	LINK1
MNCHN3	ASTSUB HDNSUB RCSUB RGISUB SQRT TRAN	LINK1
MNCHN4	AIBST SETPH AIBSUB SQRT RBSTSB XLIN RBSUB LOOKUP SEGSUB ZEROOV IBMOUT NSCE LESSQ VARI	LINK2
MDTSB	TISUB XLIN	IBMOUT
MSISUB	SQRT	SCTOR1
MTISUB	SQRT	S2SK
NSCE	ACOS SUBSON MACH XLIN RBSUB SQRT	MNCHN4 LINK2
PCWRT		LINK1
PLNCNS	ACOS SQRT ASIN	MNCHN1

6.1.2

Subroutine Linkage Table (Continued)

<u>NAME</u>	<u>LOWER-LEVEL CALLS</u>	<u>CALLED BY</u>
PLNLCS		MNCHN1
P0SUB	SQRT	ASUBC BSUBC HAPSBC HASUBC HBPSBC HBSUBC SCT0R1
PT1AA	HESUB SD1D13 SQRT ZER0DV	ASTSUB LPDAPS MNCHN2 RGISUB
P1SUB	AC0S SQRT	ASUBC BSUBC HAPSBC HASUBC HBPSBC HBSUBC
P3SUB	AC0S SQRT TDGRE	ASUBC BSIBC HAPSBC HASUBC HBPSBC HBSUBC
RASUB	AC0S SQRT	AESUB SD1D13
RASUBB	AC0S SQRT	ASUBC BSUBC HAPSBC HASUBC HBPSBC HBSUBC SCT0R1
RBSTSB	XLIN GETDAT BIAS	RBSUB MNCHN4

6.1.2

Subroutine Linkage Table (Continued)

<u>NAME</u>	<u>LOWER-LEVEL CALLS</u>	<u>CALLED BY</u>
RBSUB	RBSTSB XLIN	MNCHN4 SEGSUB SLØT NSCE
RBVSUB		MNCHN1
RCSUB	FDGRE TDGRE ZERØDV	ASESUB MNCHN2 MNCHN3
RGISUB	PT1AA SQRT	MNCHN3
RØE1SB		SCTØR1 S2SK
RØPSB		SCI
SCI	ASUBC BRAKSB BSUBC DPRASB HAPSBC HASUBC HBPSBC	HBSUBC RØPSB SQRT TRAN VSTRSB ZERØDV HDNSUB
SCTØR1	AIGSUB LBSUB MSISUB PØSUB RASUBB	RØE1SB SQRT YPSUB ZISUB HDNSUB
SCTØR2	S2SK	HDNSUB

USE FOR TYPEWRITTEN MATERIAL ONLY

6.1.2

Subroutine Linkage Table (Continued)

<u>NAME</u>	<u>LOWER-LEVEL CALLS</u>	<u>CALLED BY</u>
SDID13	ACØS HESUB RASUB SQRT XRSUB ZERØDV	PTIAA
SEGSUB	AIBSUB LPTØ RBSUB SLØT SQRT XLIN	MNCHN4
SETPH	CØNV SQRT VFPPSB	MNCHN4
SLØT	RBSUB SQRT ZERØDV	SEGSUB

SQRT

ACØS	LPTØ	SDID13
AESUB	MNCHN3	SEGSUB
AFPSUB	MNCHN4	SETPH
AIBST	MSISUB	SLØT
AIBSUB	MTISUB	SUBSØN
AIGSUB	NSCE	S2SK
ALRSUB	PLNCNS	TDGRE
ARSSUB	PØSUB	THETAR
AWESUB	PTIAA	VØLSUB
BRAKSB	PI SUB	VSEC
DPRASB	P3SUB	VSTRSB
ENDCSB	RASUB	XRSUB
FDGRE	RASUBB	SPSUBB
GAMA2S	RGISUB	SRTHR
HESUB	SCI	YPSUB
INPUT	SCTØRI	

USE FOR TYPEWRITTEN MATERIAL ONLY

6.1.2

Subroutine Linkage Table (Continued)

<u>NAME</u>	<u>LOWER-LEVEL CALLS</u>	<u>CALLED BY</u>
SUBSON	MACH SQRT	NSCE
S2SK	LBSUB MTISUB RØEISB SQRT XRTHR YPSUB ZISUB	SCTØR2
TØGRE	SQRT	P3SUB RCSUB ZISUB
THETAR	ACØS SQRT	ASUBC BSUBC HAPSBC HASUBC HBPSBC HBSUBC
TISUB		MØDTSB
TRAN		ASUBC BSUBC HAPSBC HASUBC HBPSBC HBSUBC LPTØ MNCHN2 MNCHN3 SCI
VFPPSB		SETPH
VØLSUB	SQRT VSEC	HDNSUB
VSEC	SQRT XRTHR YPSUB	VØLSUB

6.1.2

Subroutine Linkage Table (Continued)

<u>NAME</u>	<u>LOWER-LEVEL CALLS</u>	<u>CALLED BY</u>
VSTRSB	SQRT	SCI
XLIN		AIBST MNCHN4 MODTSB NSCE RBSTSB RBSUB SEGSUB
XRSUB	SQRT	AESUB SDID13
XRSUBB	SQRT	ASUBC BSUBC HAPSBC HASUBC HBPSBC HBSUBC XRTHR
XRTHR	ACOS SQRT XRSUBB	S2SK VSEC
YPSUB	ACOS SQRT	SCTØR1 S2SK VSEC
ZERØDV		AFPSUB AIBST PTIAA RCSUB SCI SDID13 SLØT
ZISUB	TDGRE	SCTØR1 S2SK

USE FOR TYPEWRITTEN MATERIAL ONLY

REV LTR

BOEING

NO. D2-125286-1 R

SH.

170

6.2

Flow Charts

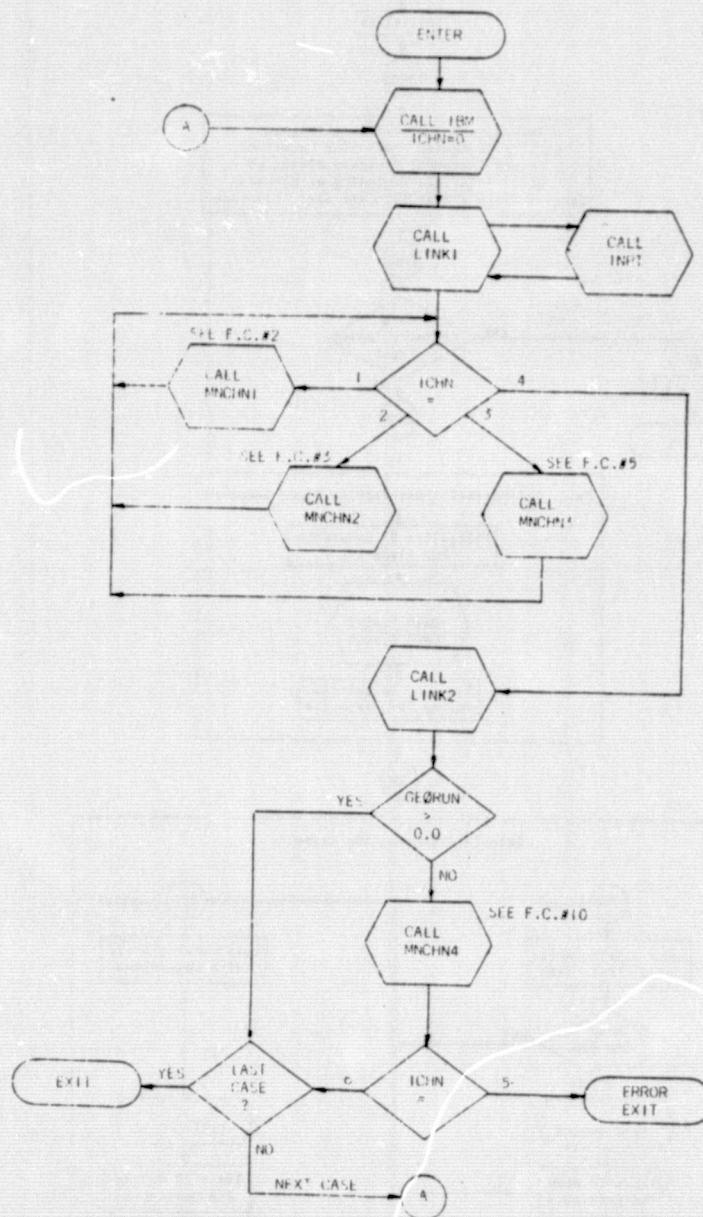
This section contains macro-scopic flow charts in schematic form of the first, second and third level subroutines. The Fortran listings themselves are considered to be the flow charts for the lower level subroutines.

These routines are annotated by comment cards which define the computational blocks and locate important logic branches.

The following schematic flow charts are included:

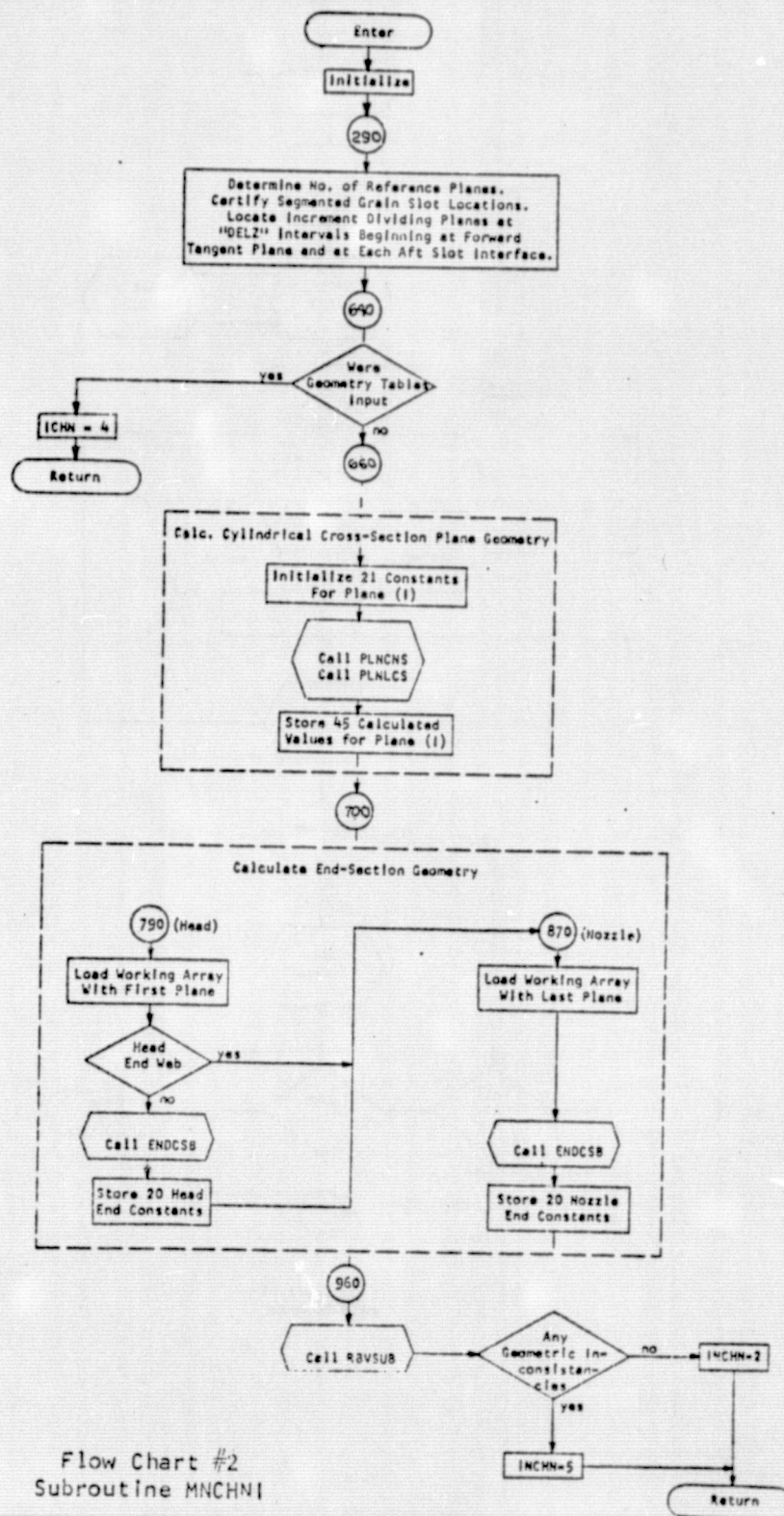
<u>Name</u>	<u>Flow Chart No.</u>
MAIN Program	1
Subroutine MNCHN1	2
Subroutine MNCHN2	3
Compute Plane Constants	4
Subroutine AESUB	5
Subroutine MNCHN3	6
Subroutine SCI	7
Subroutine SCTOR1	8
Subroutine S2SK	9
Subroutine MNCHN4	10
Subroutine SEGSUB	11
Subroutine SETPH	12
Subroutine TISUB	13

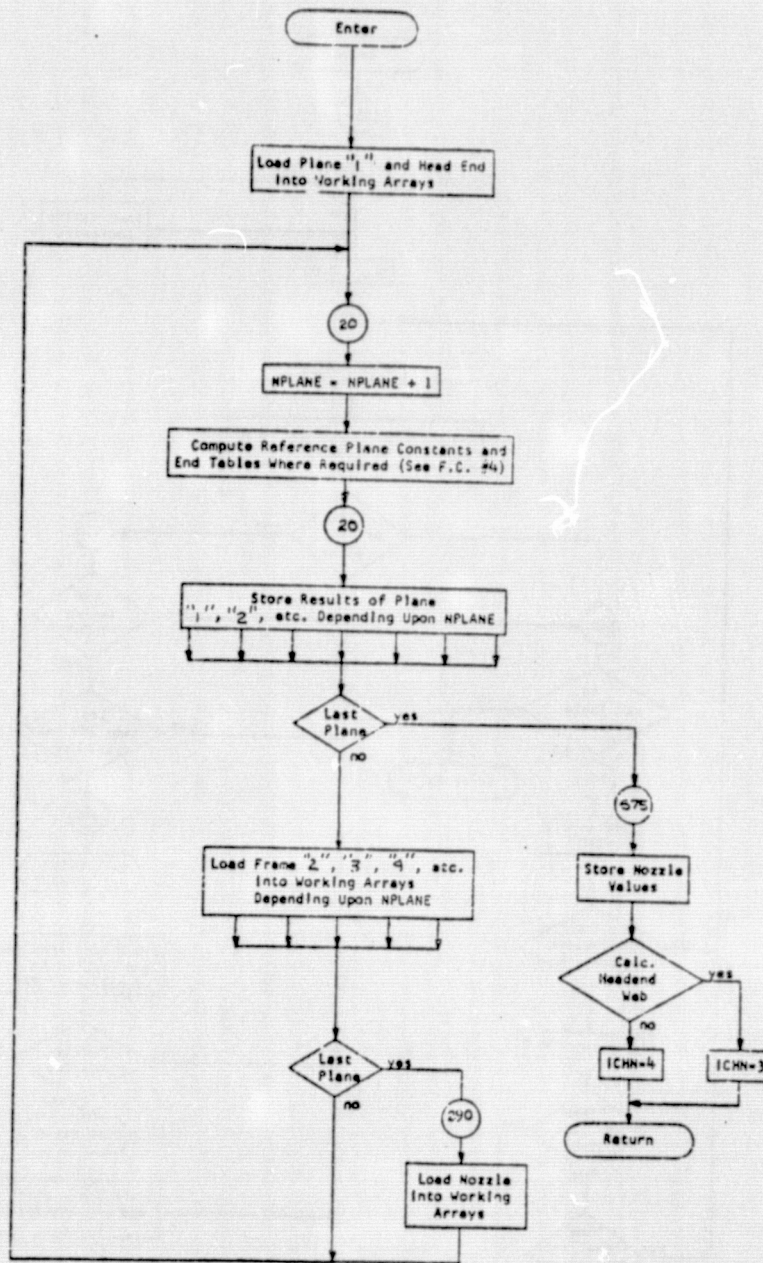
ORIGINAL PAGE IS
OF POOR QUALITY



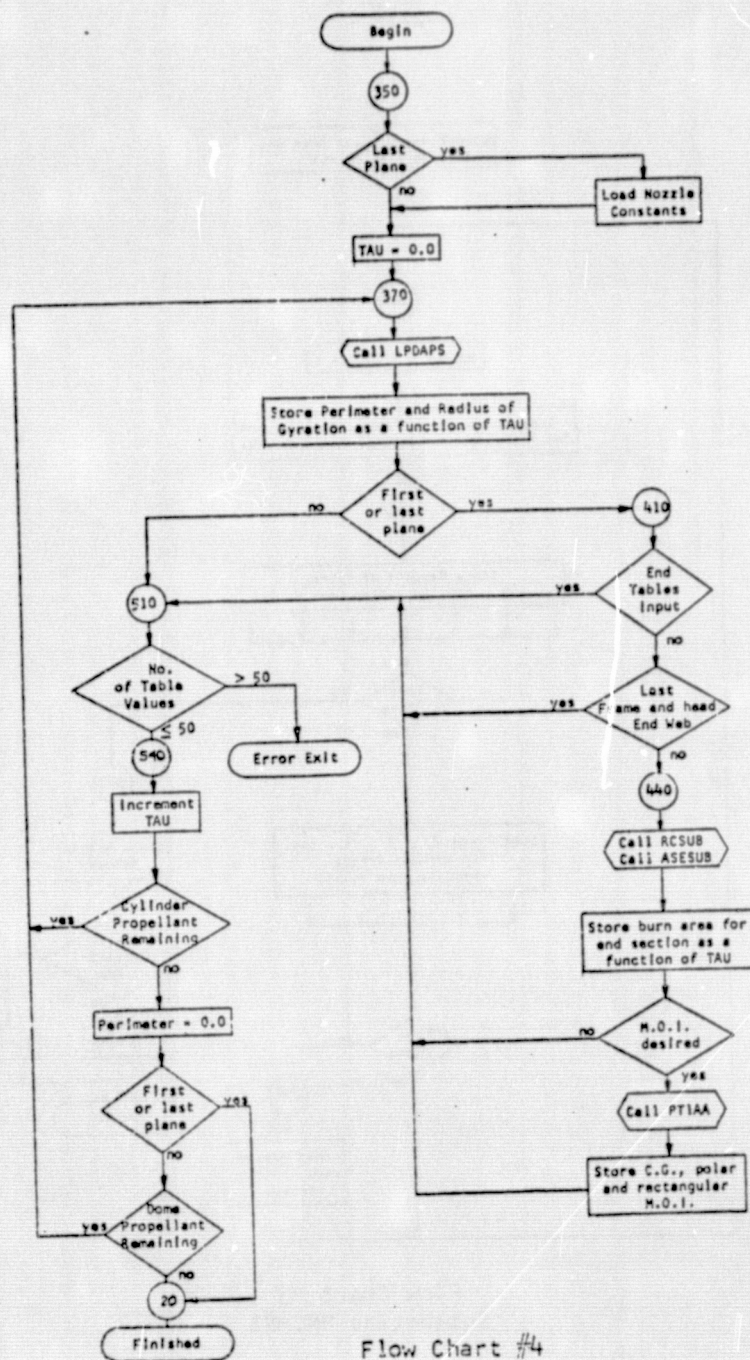
Flow Chart #1
MAIN Program

USE FOR TYPEWRITTEN MATERIAL ONLY

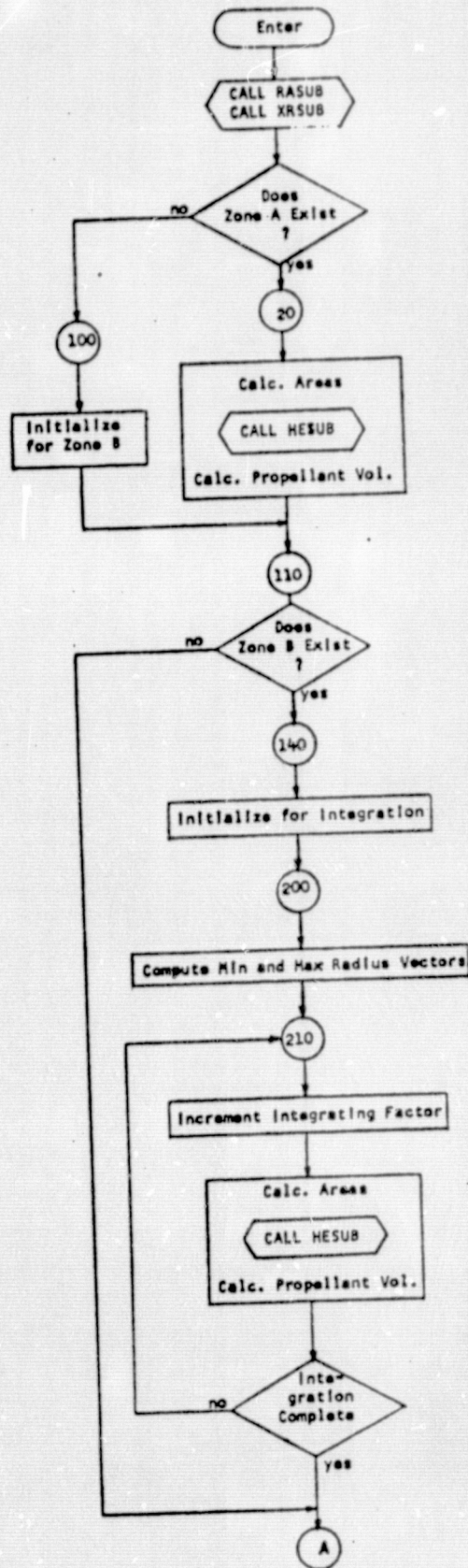




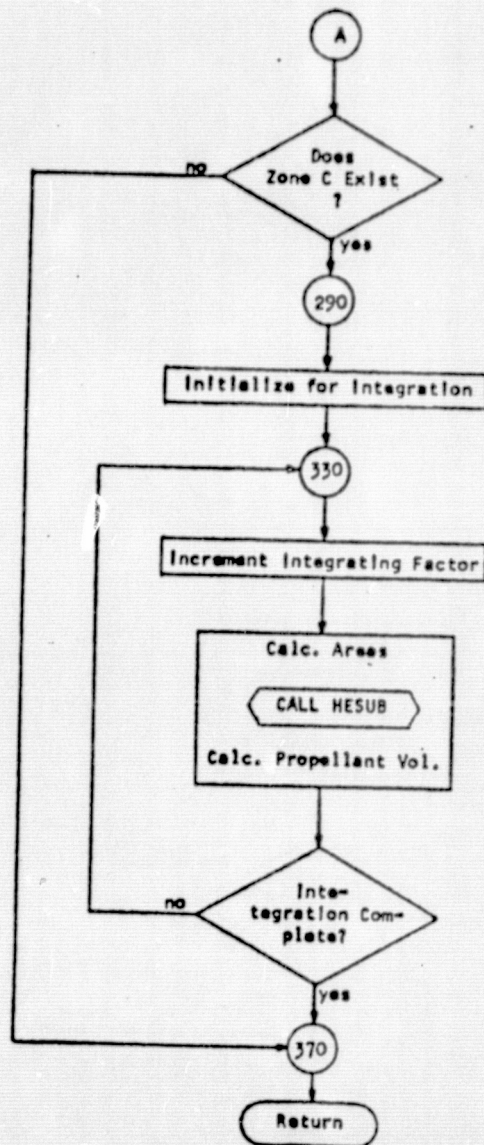
Flow Chart #3
Subroutine MNCHN2



Flow Chart #4
Compute Plane Constants

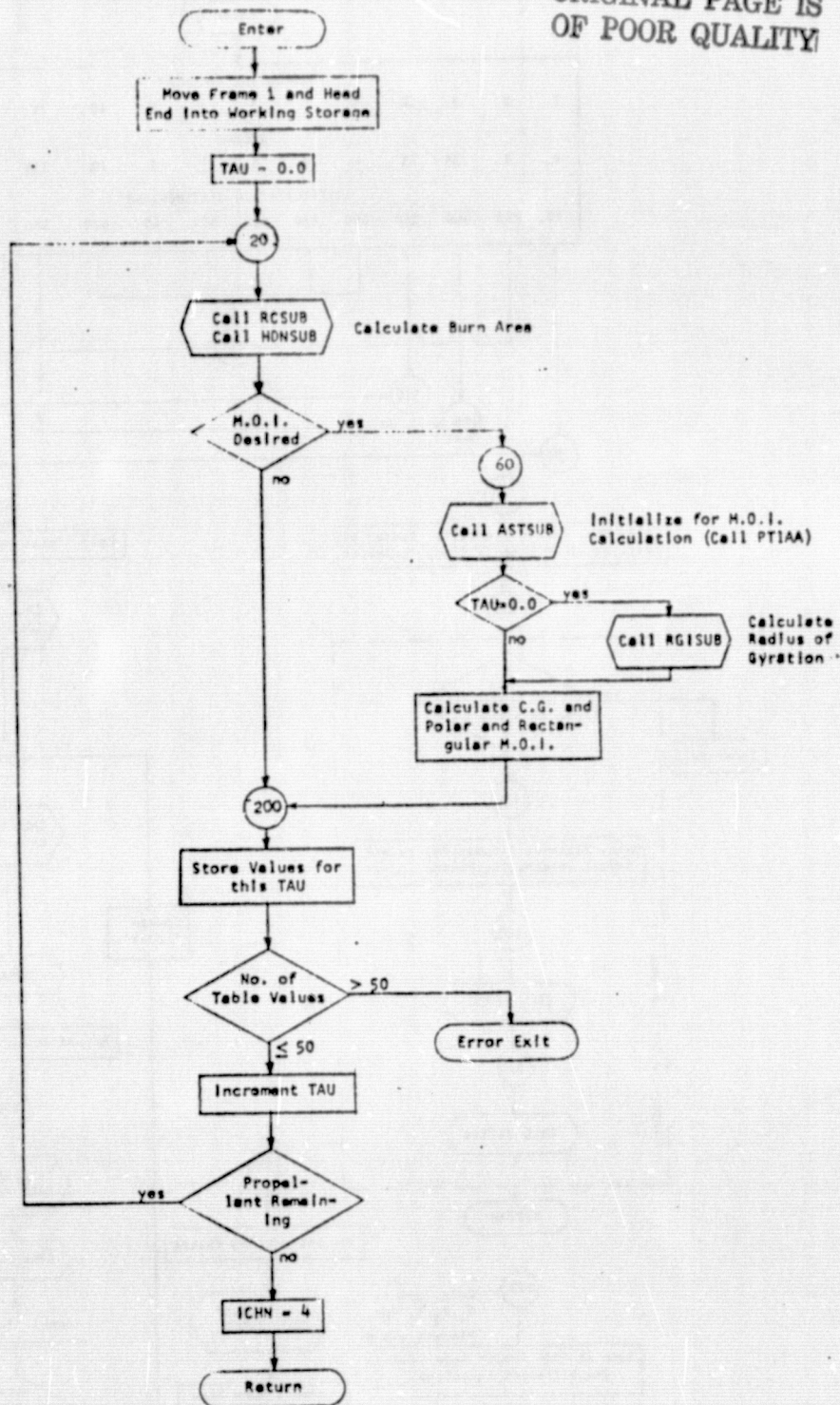


Flow Chart #5
Subroutine AESUB (1 of 2)

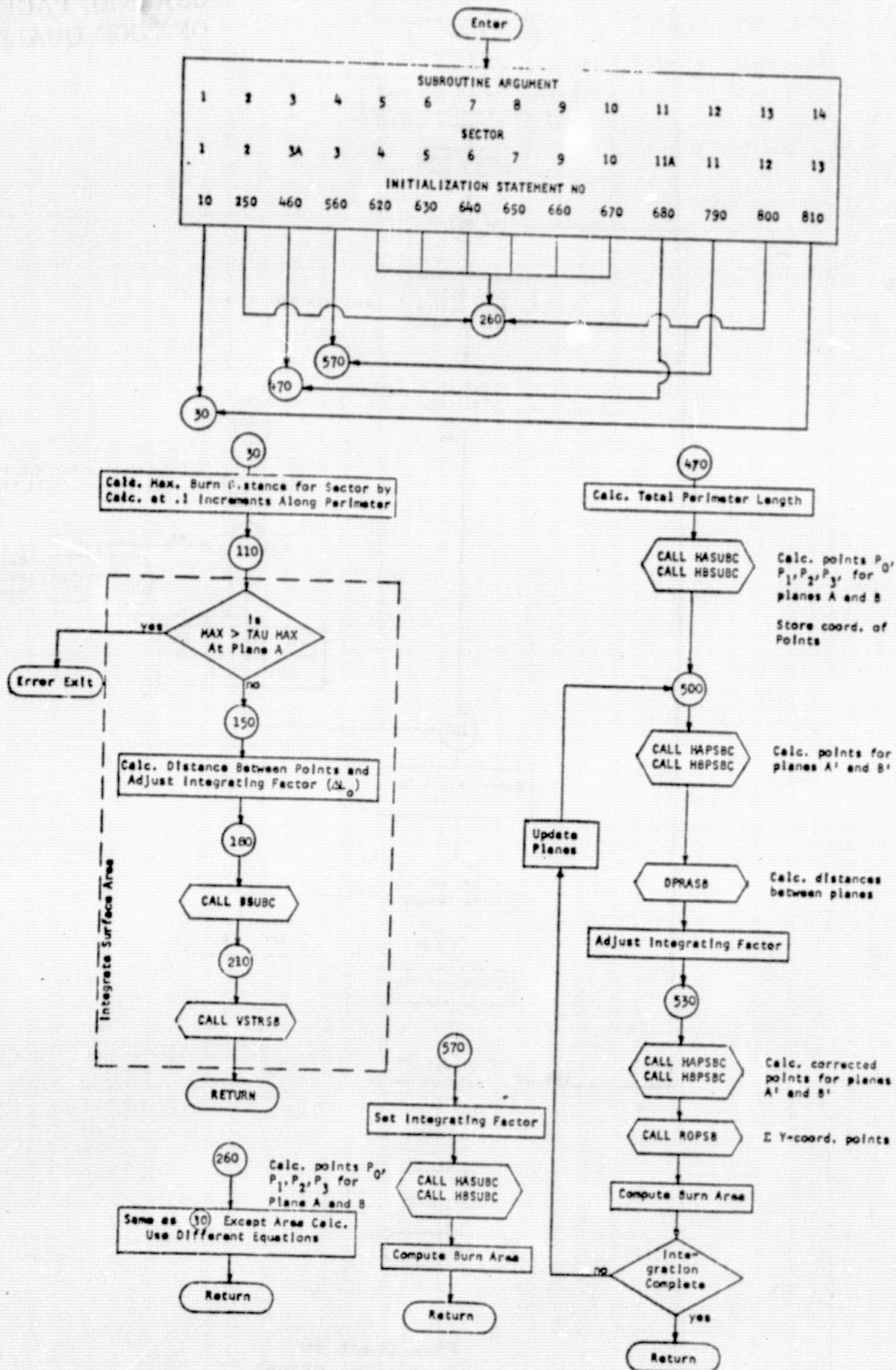


Flow Chart #5
Subroutine AESUB (2 of 2)

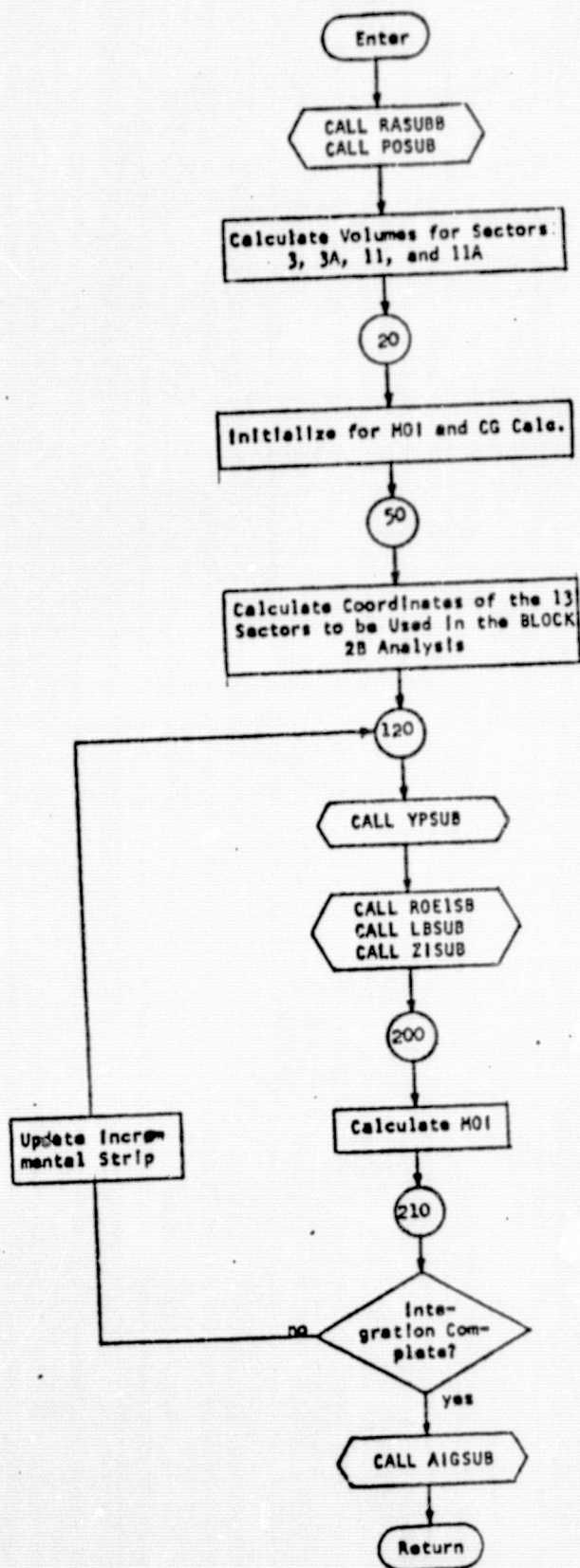
ORIGINAL PAGE IS
OF POOR QUALITY



Flow Chart #6
Subroutine MNCHN3



Flow Chart #7
Subroutine SCI



Calculate Coordinates of
Points A and B for Sectors
3, 3A, 11 and 11A

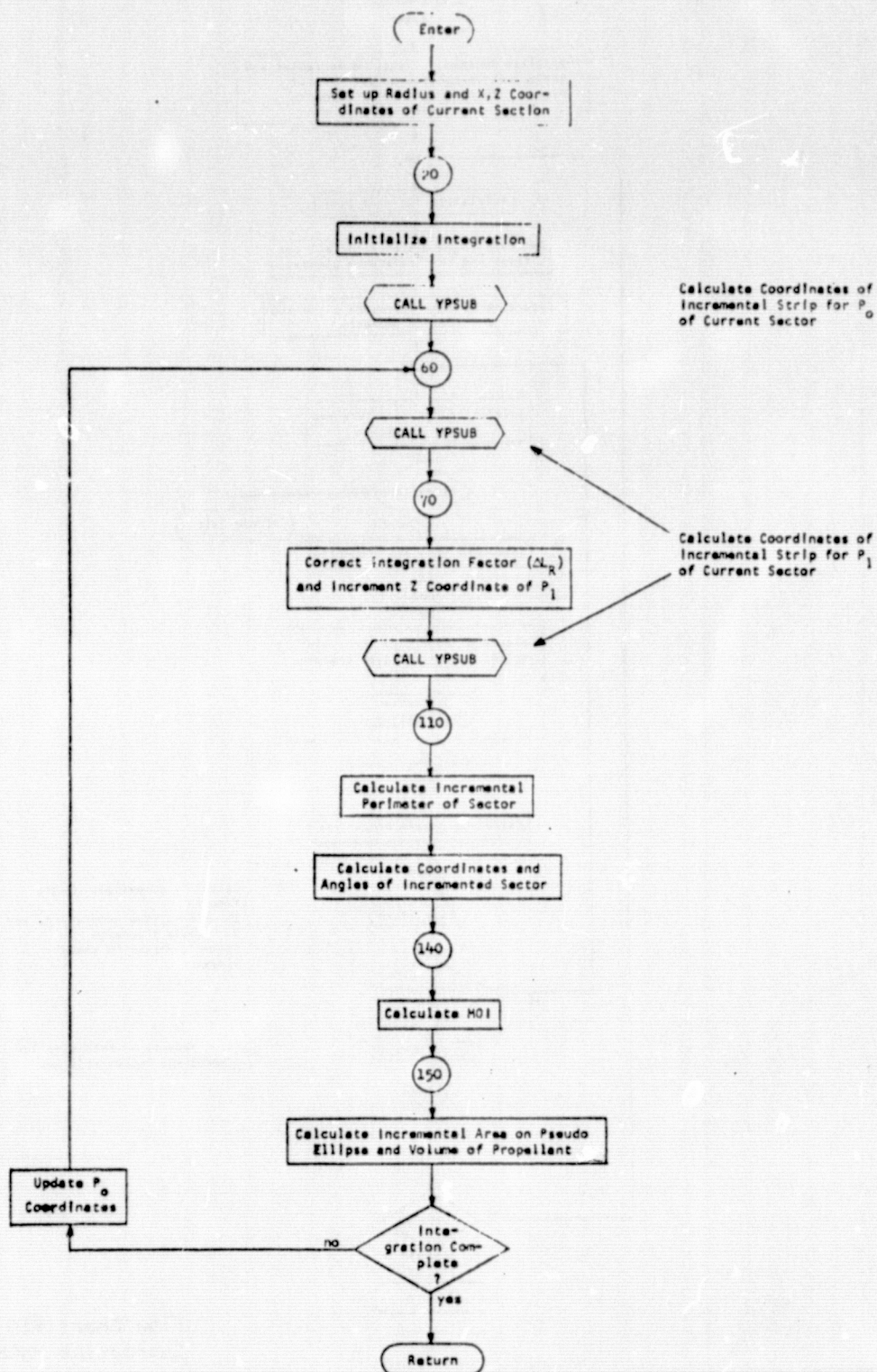
(Remaining Volumes are
Calc. in Subroutine SC1)

Calculate Coordinates of the
Incremental Strip to be Used
in Integrating for Area

Calculate Radius of Curvature
of the Incremental arc.
Calculate Y-Axis intercepts
of Lines Normal to the Ellipse.
Calculate Z Coordinates of P₁
Normal Line on Outer Ellipse.

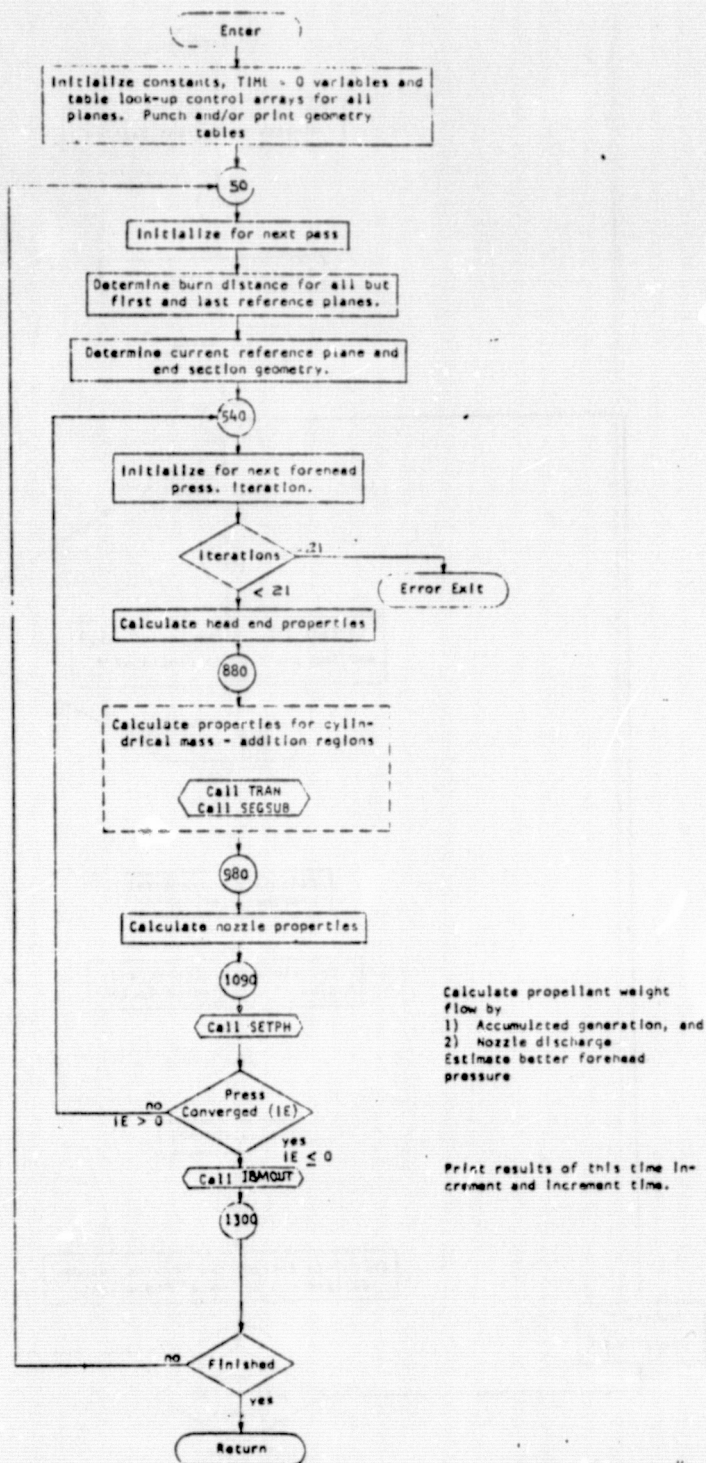
Calc. Surface Area of the
Ignitor Hole.

Flow Chart #8
Subroutine SCTØR1



Flow Chart #9
Subroutine S2SK

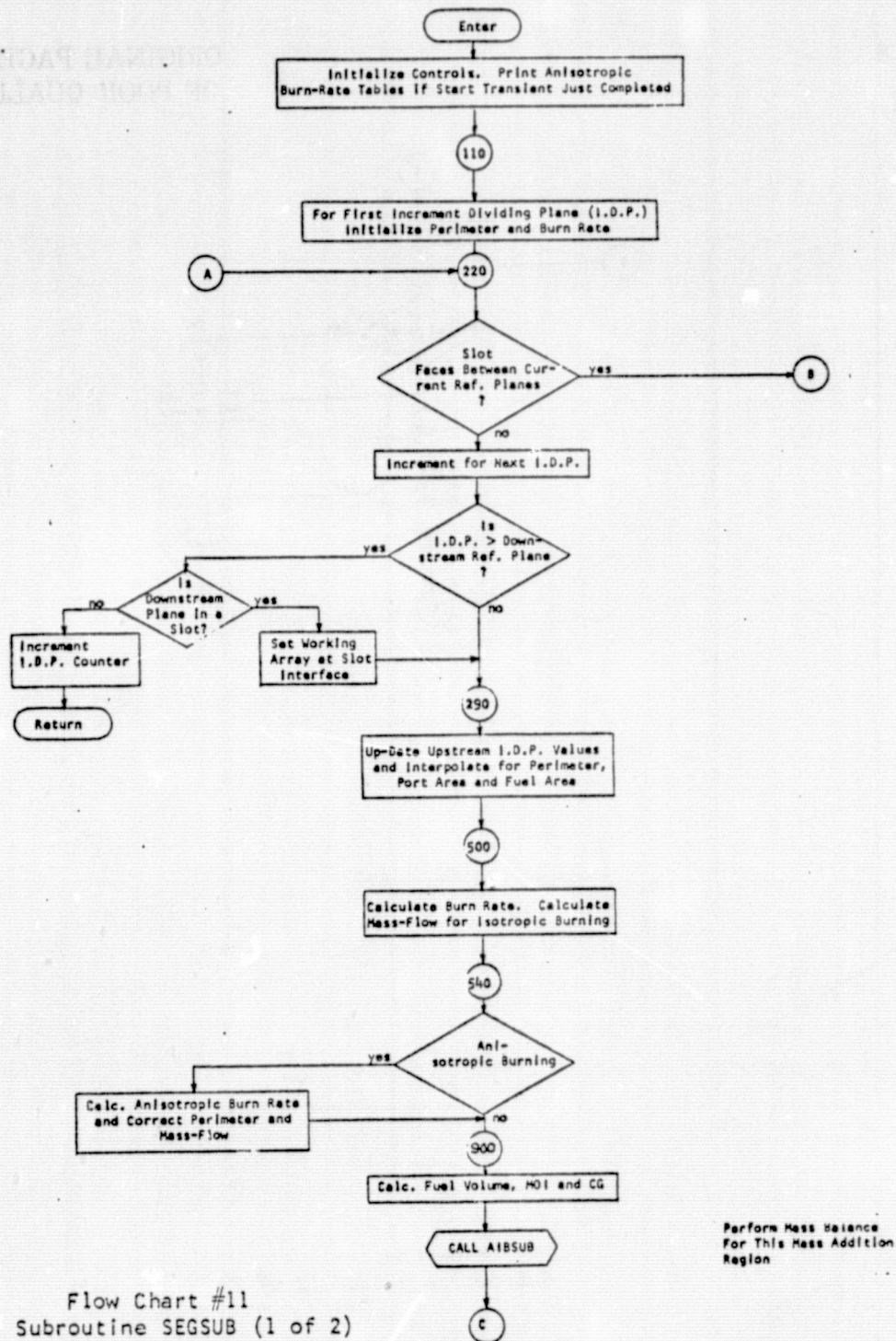
USE FOR TYPEWRITTEN MATERIAL ONLY



Flow Chart #10
Subroutine MNCHN4

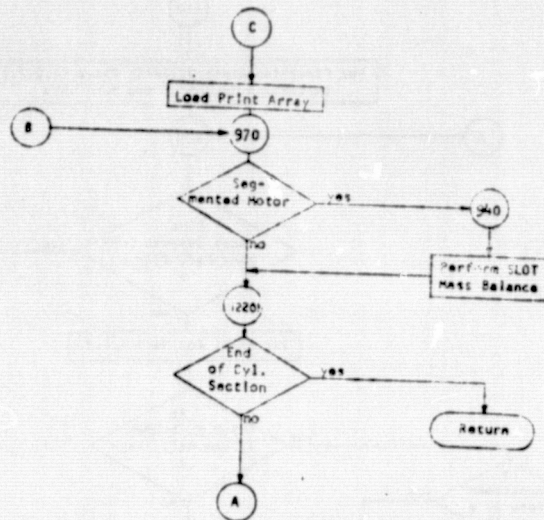
USE FOR TYPEWRITTEN MATERIAL ONLY

Flow Chart #11
Subroutine SEGSUB (1 of 2)

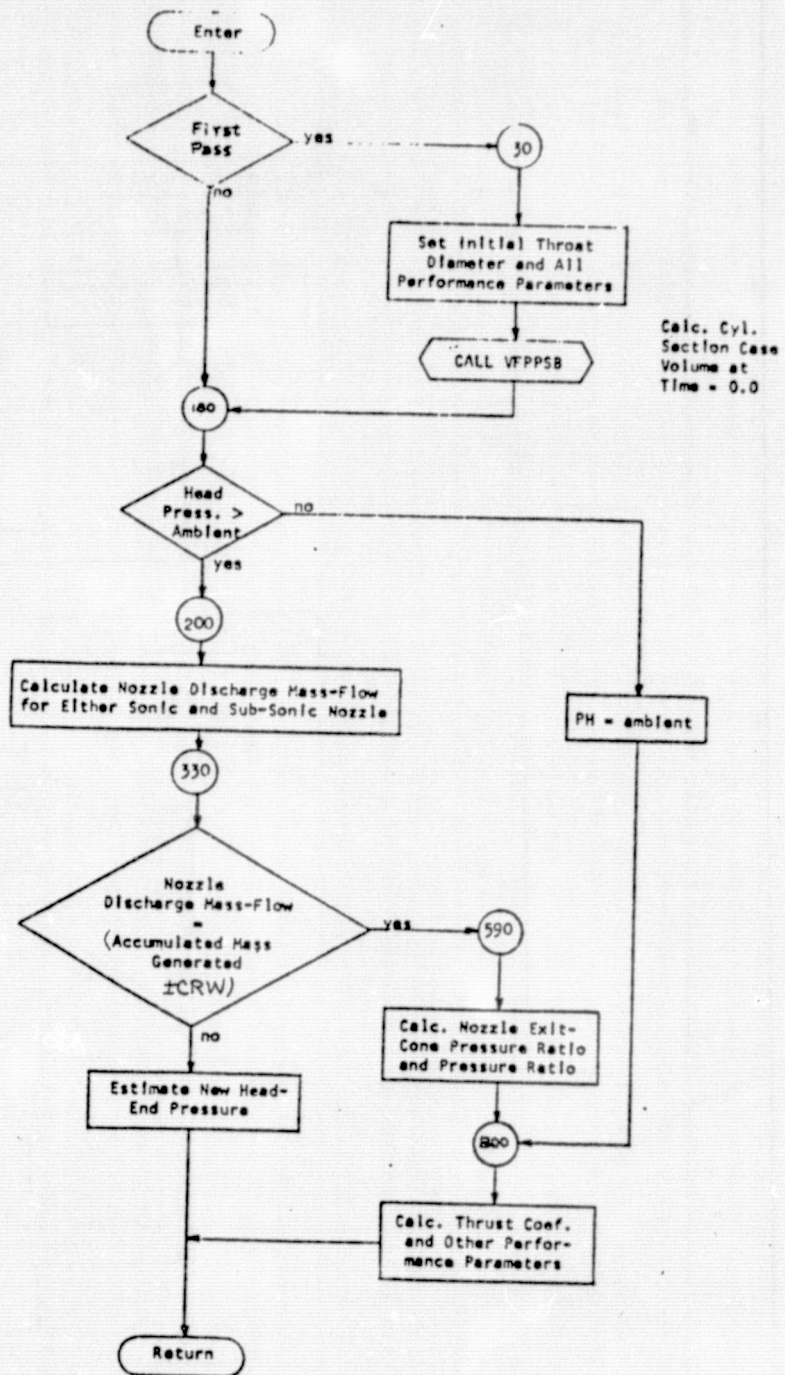


USE FOR TYPEWRITTEN MATERIAL ONLY

ORIGINAL PAGE IS
OF POOR QUALITY

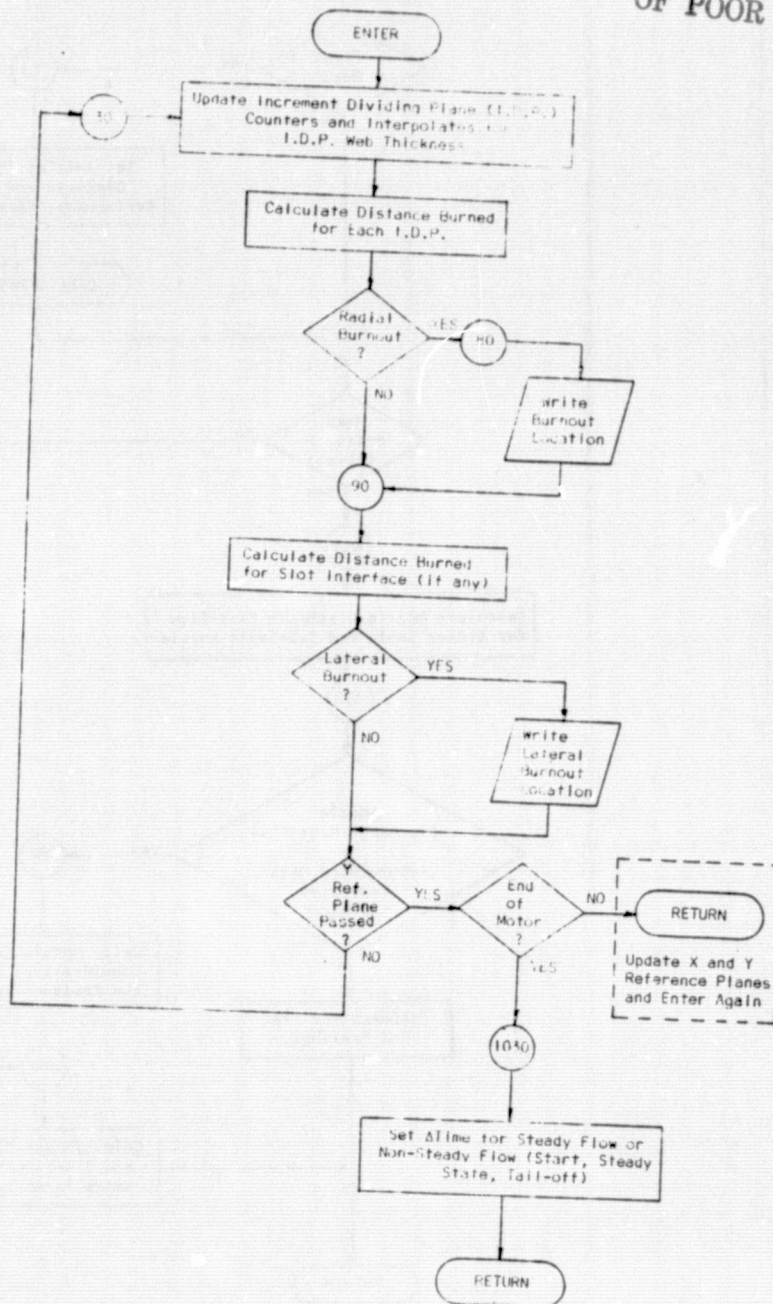


Flow Chart #11
Subroutine SEG SUB (2 of 2)



Flow Chart #12
Subroutine SETPH

ORIGINAL PAGE IS
OF POOR QUALITY



Flow Chart #13
Subroutine TISUB

**Multiple Pages Missing from Available
Version**

6.3. COMMON Allocation

The following pages contain the internal Ballistics Module COMMON Allocation. These COMMON blocks replace much of the equivalent and related COM(1) and COM(2) arrays used originally in the Internal Ballistics Program.

```

COMMON/CONSTS/PI,PI2,RADIAN
COMMON/INPUT1/ANINCIN(18),ANO(18),RF(18),FAUN(18),R2(18),
1      R3(18),R4(18),R5(18),R6(18),R7(18),R8(18),
2      ALS1(18),ALS2(18),ALC1(18),ALH(18),ALE(18),A01(18),
3      A02(18),A03(18),A04(18),A05(18)
COMMON/INPUT2/THO(18),FSLVR(18)
COMMON/INPUT3/GEJCON(45,18)
COMMON/INPUT4/NGEO(18),APORT(18),TAUPL(50,18),ALPPL(50,18),
1      AK,AY-(50,18),TAUHD(50),ABHD(50),PHOHD(50),
2      RMO,HD(50),XGHD(50),TAUN(50),XG(50),PHOIN(50),
3      RMOIN(50),XGON(50),NGEHD,NGEON
COMMON/INPUT5/HTADE,OH1,HH,AUHM,HIG,HHF
COMMON/INPUT6/CONM,HN,ONI
COMMON/INPUT7/CK,AKK,DLRF,URVRF
COMMON/INPUT8/AKKH,AKKH,HBFLAO
COMMON/INPUT9/CKG(5),AKU(5),AKK(34),AKSLUT(2),NAKR,TIMAKR(25),
1      TBLAKR(25),NAKEND
COMMON/INPUT10/PH,PHST(70),TIMEPH(70),AKRS(1),TAUAKR(30),
1      AKHTAU(30),NTAUTU,PCIAH
COMMON/INPUT11/USTAR,GAMA,K,NCSTR,PRESS(20),CSTR(20),GAMAG(20),
1      AMWG(20),TCUMB(20),NCSCOE
COMMON/INPUT12/ANCP,VCINP,VCINP
COMMON/INPUT13/ANZ,CM,DE,DT,ANN
COMMON/INPUT14/NDT,TIMEOT(25),TDELDT(25),ERHAR,EREXP,PONBAR
COMMON/INPUT15/NCM,TCLCM(50),TIMECM(50),TIMAX,CKDUMP(80),NPA,
1      TIMEPA(50),TSLPA(50)
COMMON/INPUT16/PRIFLG,NWR,FAB,INPUTAB
COMMON/INPUT17/STFLAG,STOYST,DELTSI,DELTS,DELTT0
COMMON/INPUT18/ATST,PST,IST,TIMPT1,TIMPT2,DELTS,ANITW
COMMON/INPUT19/KPLANE,KMOICG
COMMON/INPUT20/NEPS,TIMEPS(50),EPGA(50),EPCN(50),NACCEL,ACCEL(50),
1      TIMEAC(50)
COMMON/INPUT21/CRP,CRF,CRW
COMMON/INPUT22/FACT,PHOTOL,PTOL
COMMON/INPUT23/DTAU(18),DTAUW(18)
COMMON/INPUT24/SB(18),SLFBRN(18),SA(18)
COMMON/INPUT25/DEL,PA,PHI,FCO,DELZ,KDUMP(72)
COMMON/INPUT26/VFNOA,VFNOH,VCNINA,VCNINH
COMMON/INPUT27/TAUMNA,TAUMNB
COMMON/INPUT28/A(5),R1,R9,TH1,T2M,T4M,T5M,X45,Y45,ALC,XU3,YU3,
1      RU3,XU5,YU5,RU5,X)/,YU7,RU7,T6M,T7M,T12M,X76,Y76,
2      ALD,XU9,YU9,RU9,XU11,YU11,RU11,T10M,T9M,TAUM,TH2,
3      TH3,TH4,B71M,B72M,B91M,B92M
COMMON/WORKX/AINC(1),ANOUM,RFDM,TAUWDM,R2DM,R3DM,R4DM,R5DM,R6DM,
1      R7DM,R8DM,ALS1DM,ALS2DM,ALADM,ALBDM,ALEDM,AW(5)
COMMON/WORKDE/DE1,BE,AUEM
COMMON/WORKRE/RE1(1),ALFEM,ALFE,RE2,HE1,HE2,HEO,HER,VFE0,VCE,
1      TAU0,TAUE1,CAE,CBE,CCCE,CCVE,CDCE,CDVE,CECE,CEVE
COMMON/WORKRH/RH1(5),HH2,ACG,HHW,VFHO,VCH,ANK(10)
COMMON/WORKRN/RN1(5),VFNO,VCN,ACK(10)
COMMON/XOAPW/RAAP,XRATAP,THRAP,GAM1AP,GAM2AP,XOAP,YOAP,ZOAP,
1      X2AP,Y2AP,Z2AP,X1AP,Y1AP,Z1AP,UAP(4),TG2AP,STRAP,
2      CTRAP,AAP,BAP,CAP,UAP,BTAUAP,SOAP(4),ALTAP
COMMON/XOBBW/XABP,XRATBP,THRBP,GAM1BP,GAM2BP,XOBP,YOBP,ZOBP,
1      X2BP,Y2BP,Z2BP,X1BP,Y1BP,Z1BP,UBP(3),TG2BP,STRBP,
2      CTRBP,TP1BP,ABP,BBP,CBP,UBP,BTAUBP,SOBP(4),ALTBP

```

6.3.

COMMON Allocation (Continued)

```

COMMON/XUAWOR/KAA,XKATA,THKA,GAMA1A,GAMA2A,XOA,YOA,ZOA,X2A,Y2A,
1      Z2A,X1A,Y1A,Z1A,X3A,Y3A,Z3A,TANG2A,SINTRA,CUSTRA,
2      TANP1A,AAAA,BA,CA,DA,BTAUA,SINGA2,COSGA2,SINGA1,
3      COSGA1,ALTA
COMMON/XUBWOR/KAB,XKATB,THKB,GAMA1B,GAMA2B,XOB,YOB,ZOB,X2B,Y2B,
1      Z2B,X1B,Y1B,Z1B,X3B,Y3B,Z3B,TANG2B,SINTRB,CUSTRB,
2      TANP1B,ABB,BB,CB,DB,BTAUB,SINGB2,COSGB2,SINGB1,
3      COSGB1,ALTB
COMMON/XUWOK/XO,YO,ZO,X2,Y2,Z2,X1,Y1,Z1,X3,Y3,Z3,TANGM2,SINTHR,
1      CUSTHK,TANPH1,AU,BU,CU,DU,BTAO,SINGM2,COSGM2,
2      SINGM1,COSGM1,ALTO
COMMON/CUMA/DEL1,APH1,WOUT1,ANIBU,TIME*,UT,ANLOPS,ACCEL,
1      ABCYL,PKNT(101,15),A1NCHI,AMACH,ZCALC(101)
COMMON/CUMB/AMH,AJPHN,AJBHN,TAUUA,AJPHD,AJBHD,AJPNZ,AJBNOZ,
1      XBH,XBN,DTINT
COMMON/CUMC/AMTJ,AMTI,AJPHU,AJBHU,XBIH,ALQ
COMMON/CUMD/BHOLD,KHOLD,RGF,RGI,XBARST,AJSTP,AJSTB,WSTST,
1      KGF,B,RGI
COMMON/CUME/TAUWEI,XCGA,XCGH,ZHU,ACGA,ACGB,RCGO,ZCG,YCG,YHO,YI
COMMON/CUMF/AMSI,AMSI
COMMON/CUMG/TAUZ(101),RBZ(101),TAUZTO(101),RBZTO(101),PD(101),
1      TAUWUP(101),RB,VF,UWUOT,VP
COMMON/CUMH/PDPR(101),V(101),VPR(101)
COMMON/CUMI/ALIVAC,FVAC,VFCYL,CPPROP,FINDP
COMMON/CUMJ/ABSLTA(18),ABSLTF(18),APA(18),APF(18),PSA(18),
1      PSF(18),PUA(18),POF(18),TSA(18),TSF(18),UA(18),
2      UF(18),WSLUT(18),WSLUT1(18)
COMMON/CUMK/DWSLTA(18),DWSLTF(18),UWUTS(18),AFH1,RBSLTA(18),
1      RBSLTF(18),TSLUTA(18),TSLUTF(18)
COMMON/CUML/THU
COMMON/CUMM/TAUTUZ,RSLVRN,AX(45),AY(45),A1NCX,ANOX,RFX,TAUWX,
1      DUMX(17),A1NCY,ANOY,HFY,TAUWY,DUMY(17),VFPP
COMMON/CUMN/TSLVX,ISLVY
COMMON/CUMO/THSLV,TSLVUM,THSLVV,ASLVR,AMU,RFH1,ALPX,APX,
1      RBH1,ALPY,APY,VSLVM
COMMON/CUMP/DELL,RT,ALAMIN,AWE,KAMIN,ALX,ALAMDA
COMMON/CUMQ/DELLU,DELLU1
COMMON/CUMR/DELLR,KPZ,APZ,AZ
COMMON/CUMS/TPK,AKKAUJ,IEND
COMMON/CUMT/TAU,HC,SUMOV,XH,HE,AL(13),XBARIH,ASE,AFF,WI,WI,RA,
1      KAO,ALL,AJPP,ASI
COMMON/CUMU/VRA,ALITU,COUNT,VFHEW1,VEX,VR,ASTO,ABIGR
COMMON/CUMV/VSR,VSTO,TUMAX,DO1A,DO1B,XMAX,ZMAX,YMAX,ALHO,
1      ROPE1,ROPE2,ROPE3,ALDP
COMMON/CUMW/DV,AEE,PEPU1
COMMON/CUMX/CUSA(5),SINA(5),DTAUX,DTAUXX
COMMON/CUMY/AV1,AV2,TV2,TV4,TV5,TV6,TV7,ALVA,ALVB,ALVC,RV2,RV3,
1      RV4,XV7,DELL3,RV1
COMMON/CUMZ/AV3,BV1M,BV2M,THV,RV5,RV7,BV1,BV2,RSLVRV
COMMON/PARMA/AFI,AL7,AL8,TAUTOV,BVX,BVXX
COMMON/PARMB/AP,PMIN,PMAX,WOUT,111,11J,WOUTD,NSLOT,NTABE,NTME,
1      TAUTO,TUFLAG,N1NCPL,BRNOUT,11S,1S1,1S2,N1,SCUR(18,2)
COMMON/PARMC/DEL3,DEL7,B71,B71
COMMON/PARMD/KRASUB,KASUB,AJBH,HE1,AJBH,AJBH,XBAR1
COMMON/PARME/RCG,DELLI,XBARIN,HCK,KWIT1,KWIT2,LSWIT1,ALSX,
1      AJPN,AJPH

```


6.3. COMMON Allocation (Continued)

```

COMMON/PARMF/PM,TIME,AINC,T,P,DELTA,U,AKGY,PON,DIS,AMPN,AT,
1 AM,AKRST
COMMON/PARMG/VFH,AAN,VFN,VIS,AIT,SPHDT,SPONDT,VFINI,VEH,ABTOI
COMMON/PARMH/BIE,BUE,AANN,BX,RXX,ASI1,DELLI,HOPE4,AIE,YPI,ZPI,
1 AKU,AKC1,HOE1,ALITL,ZI,AIG,THH1,THRO,AUE
COMMON/PARMI/AUMCYL,AIBCYL,AIPCYL,AKGYX,AKGY,YPH,VPN
COMMON/PARMJ/ETH
COMMON/PARMK/F,EP1,PEPU,CFUL,VFWE,WD,DEED,CLUPS,CFO,WGTOT,
1 SWOUTN
COMMON/PARML/HOLDH,AL3A,BRAK,AL11A,AS,RPX,ZPO,YPO,DS,KBRAK,KVSTR
COMMON/PARMM/RAT,XRAT,THR,GAMA1,GAMA2,ZLAT
COMMON/PARMN/UPRA,UPSA
COMMON/PARMO/ALP,KRASBB,KARSBB,KGAM
COMMON/PARMP/THH2,RP1,RP2,RP3,RP4,RP5,RP6,RP10,RP11,RP12,RP13,XP1,
1 XP3,XP5,XP11,XP13
COMMON/PARMQ/KGYF(18),AFRPL(18),ALPRP(18)
COMMON/PARMR/RBSLOT,NSLUTA,NSLUTF,SLTFEG
COMMON/PARMS/ICHN
COMMON/PARMI/ANI
COMMON/PARX/IVWF
COMMON/PARMW/WGPORT
COMMON/PARMX/WF1,EPG
COMMON/PARMY/PSAPR(18),PSFPR(18)
COMMON/PARMZ/ABSLUT,PU
COMMON/PARMAA/TU,IFLAG
COMMON/PARMAB/NSUBMG,NSUBMG,NEND,ASEA,ASEB,SUMOVA,SUMOVH
COMMON/PARMAC/HEA,HEB,AEEA,AEEB,DVA,DVB,HEIA,HEIB,AWEA,AWEB,DELH
COMMON/PARMAD/PMOINA(50),PMOINB(50),RMOINA(50),RMOINB(50),
3 XCGNA(50),XCGNB(50),ABNA(50),ABNB(50)
COMMON/PARMAE/XBARIA,XBARIB,AJPPA,AJPPB,AJBBA,AJBBS
COMMON/PARMAF/WIA,WIB,WTAW,WTB
COMMON/PARMAG/TXSUB(20),TXSUB(20),NPSUB
COMMON/PARMAH/VNOZSB,THETEX,SIRATC,NCONT
COMMON/PARMAI/VFNA,VFNB,VCNA,VCNB
COMMON/PARMAJ/AANA,AANB,AJPNZA,AJPNZB,AJBNZA,AJBNZB,XBNA,XBNB
COMMON/PARMAK/PIAT,AENI,U1A,PIA,PIB,AZA,VGA,PMA,PHA,TMA,VGAZ,
3 MUBETA,U1B,A1A,MU1B
COMMON/PARMAL/DWUTA,DWUTB,DWUTA,DWUTB,PIBT,TIB,MIB
COMMON/PARMAM/TA1B(50),TA2B(50),TAUNB(50)
COMMON/PARMAN/A1B,A2B
COMMON/DUMYA/ALH,ALRI,ALHMAX,ALRS,AR,GAMAT,XHO
COMMON/DUMYB/AFU
COMMON/DUMYC/T,ST,TIMECK
COMMON/DUMYD/TMP,CTMP
COMMON/DUMYG/TIMCK1,AKRHLD,PCTABA
COMMON/DUMYJ/AJHEW,AJPHW,AKRFB,AKRFP,AKRIB,AKRIP,XBARHF,XBARHI
COMMON/DUMYK/PKCRIT,APORTN
COMMON/DUMYL/TIMCK2
COMMON/DUMYM/CKTIME,NTEST4
COMMON/DUMYN/ALSUB3,ALX,DO3,DPP,DPS
COMMON/DUMYU/ASHOU,TAULST
COMMON/DUMYP/AFX,AFY,AKGYH1,ALPHI,TAUMW,THOLD
COMMON/DUMYV/ALAMN,CF,CODET,DELLP,DISB,EPS,FX,PHMAX,PHOLD,
1 PHX,PONX,PSUPU,TSS,FX,WDB,WOUTX
COMMON/DUMYS/TH,BBBB
COMMON/DUMYK/JUMP,NBACK
COMMON/DUMYT/XRI
COMMON/DUMYU/DELD
COMMON/DUMYV/GEORUN
COMMON/DUMYX/TAURPL(18)
COMMON/DUMYZ/VGBZ,PMBZ,VGB,TIMEZ,IPASS

```

Multiple Pages Missing from Available
Version

6.5

Nomenclature

The following pages contain a partial internal ballistics program nomenclature list of input and output variables, variables used in this document text, and a significant number of important, internal program variables.

ORIGINAL PAGE IS
OF POOR QUALITY

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
A	A		GENERAL FLOW AREA OF A CROSS SECTION.
	A(I)	RADIANS	WORKING ARRAY STORAGE : LOCATION OF GRAIN DESIGN GEOMETRY ANGLES.
	AAA	- - -	COEFFICIENT OF FIRST ORDER TERM IN THIRD DEGREE EQUATION IN SUBROUTINE P3SUB WHICH LOCATES THE POINT P3 ON THE OUTER ELLIPSE OF HEAD-END WITH WEB.
A _N	AAN	SQ IN	NOZZLE-END SECTION BURN AREA.
NO	AANN	- - -	TEMPORARY VALUE OF NUMBER OF GRAIN CROSS SECTION SYMMETRICAL PARTS USED IN SUBROUTINE PT1AA.
	AAP	- - -	COEFFICIENT OF FIRST ORDER TERM IN SECOND DEGREE EQUATION $X^{**2} + Y^{**2} + A*X + B*Y + D=0$ USED TO APPROXIMATE ELLIPSE BETWEEN POINTS P2 AND P3 IN THE A PRIME PLANE FOR THE BLOCK 2A ANALYSIS OF THE HEAD-END WITH WEB IN SUBROUTINE SCI.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	ABB	- - -	COEFFICIENT OF FIRST ORDER TERM IN SECOND DEGREE EQUATION $X^{**2} + Y^{**2} + A*X + B*Y + D=0$ USED TO APPROXIMATE ELLIPSE BETWEEN POINTS P2 AND P3 IN THE B PLANE FOR THE BLOCK 2A ANALYSIS OF THE HEAD-END WITH WEB IN SUBROUTINE SCI.
A_{Bcyl}	ABCYL	SQ IN	CYLINDRICAL SECTION TOTAL BURN AREA.
	ABHD	SQ IN	FORE-HEAD GEOMETRY TABLE BURN AREA. (DEPENDENT VARIABLE)
A_R	ABIGR	SQ IN	CROSS- SECTIONAL AREA OF WEB ZONE IN BLOCK 3 ANALYSIS OF HEAD-END WITH WEB. USED IN SUBROUTINE VOLSUB.
	ABN	SQ IN	AFT-HEAD GEOMETRY TABLE BURN AREA. (DEPENDENT VARIABLE).
	ABP		COEFFICIENT OF FIRST ORDER TERM IN SECOND DEGREE EQUATION $X^{**2} + Y^{**2} + A*X + B*Y + D=0$ USED TO APPROXIMATE ELLIPSE BETWEEN POINTS P1 AND P3 IN THE B PRIME PLANE FOR THE BLOCK 2A ANALYSIS OF THE HEAD-AND WITH WEB IN SUBROUTINE SCI.
A_{Bslot}	ABSLOT	SQ IN	TOTAL BURN AREA ON GRAIN SEGMENT FACES FOR SEGMENTED MOTORS.
	ABSLTA	SQ IN	BURN AREA ON SLOT AFT INTERFACE.
	ABSLTF	SQ IN	BURN AREA ON SLOT FORWARD INTERFACE.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
A_{BTot}	ABTOT	SQ IN	TOTAL MOTOR BURN AREA.
$\frac{a}{g_o}$	ACCEL	- G-S -	CURRENT TIME VALUE OF VEHICLE ACCELERATION OBTAINED FROM ACCELERATION-TIME CURVE IN SUBROUTINE AIBST.
	ACCEL T	- G-S -	VEHICLE ACCELERATION-TIME CURVE DEPENDENT VARIABLE.
	ACGA	SQ IN	SECTOR AREA USED IN SUBROUTINE MSISUB TO OBTAIN POLAR MOMENT OF CROSS SECTION FOR HEAD-END WITH WEB.
	ACGB	SQ IN	SECTOR AREA USED IN SUBROUTINE MSISUP TO OBTAIN POLAR MOMENT OF CROSS SECTION FOR HEAD-END WITH WEB.
A_{EE}	AEE	SQ IN	AREA OF SECTOR IN SUBROUTINE AESUB FOR END SECTION ANALYSIS. ALSO, NOZZLE EXIT AREA IN SETPH.
α_{ER}	AER	RADIANS	ANGLE BETWEEN MOTOR CENTER LINE AND TANGENT TO END SECTION ELLIPSE. DETERMINED IN SUBROUTINE ENDCSR.
	AFD	SQ IN	AREA OF SECTOR 3A OR 11A IN SUBROUTINE AFPSUB

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	AFF	SQ IN	CROSS SECTIONAL AREA OF INCREMENT DIVIDING PLANE IN SUBROUTINE SEGSUB.
	AFHI	SQ IN	CROSS SECTIONAL AREA OF UPSTREAM INCREMENT DIVIDING PLANE IN SUBROUTINE SEGSUB.
	AFP	SQ IN	CROSS SECTIONAL AREA OF REFERENCE PLANE IN SUBROUTINE AFPSUB.
	AFRPL(1) THRU AFRPL(18)	SQ IN	REFERENCE PLANE CROSS SECTIONAL PROPELLANT AREA IN SUBROUTINE MNCHN4. DETERMINED FROM INTEGRATION OF PERIMETER LENGTH AT EACH TIME INCREMENT.
	AFX	SQ IN	REFERENCE PLANE X CROSS SECTIONAL PROPELLANT AREA IN SUBROUTINE SEGSUB.
	AFY	SQ IN	REFERENCE PLANE Y CROSS SECTIONAL PROPELLANT AREA IN SUBROUTINE SEGSUB.
A_H	AHH	SQ IN	HEAD-END SECTION BURN AREA.
A_{H0}	AHO	SQ IN	INITIAL AND PAST TIME VALUE OF HEAD-END SECTION BURN AREA.
I_{Bcy1}	AIBCYL	SLUG-SQ IN	CYLINDRICAL SECTION RECTANGULAR MOMENT OF INERTIA DETERMINED IN SUBROUTINE SEGSUB.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
A_{IG}	AIG	SQ IN	IGNITER OPENING SURFACE AREA DETERMINED IN SUBROUTINE AIGSUB.
	AINC	IN	TEMPORARY STORAGE LOCATION OF CURRENT INCREMENT DIVIDING PLANE LOCATION MEASURED FROM FORWARD TANGENT PLANE.
	AINCIN(1) THRU AINCIN(18)	IN	REFERENCE PLANE LOCATION MEASURED FROM FORWARD TANGENT PLANE.
	AINCN	IN	NOZZLE SECTION REFERENCE PLANE OR AFT TANGENT PLAN LOCATION MEASURED FROM FORWARD TANGENT PLANE.
NINCPL	AINCPL	IN	INCREMENT PLANE LOCATION MEASURED FROM FORWARD TANGENT PLANE WHERE ANISOTROPIC BURN RATE COEFFICIENT IS TO BE EVALUATED DURING START TRANSIENT INTERVAL.
	AINCW	IN	WORKING INCREMENT DIVIDING PLANE LOCATION IN SUBROUTINE SEGSUB.
	AINCX	IN	REFERENCE PLANE X LOCATION IN SUBROUTINE SEGSUB.
	AINCY	IN	REFERENCE PLANE Y LOCATION IN SUBROUTINE SEGSUB.
I_{Pcy1}	AIPCYL	IN**4	CYLINDRICAL SECTION POLAR MOMENT OF INERTIA OF CROSS SECTION IN SUBROUTINE SEGSUB.
I_T	AIT	LBF-SEC	TOTAL IMPULSE

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
I_{Tst}	AITST	LBF-SEC	START TRANSIENT INTERVAL TOTAL IMPULSE. WHEN THIS VALUE IS INPUT, THE START TRANSIENT INTERVAL WILL BE TERMINATED WHEN THE TOTAL IMPULSE, AIT, IS GREATER THAN OR EQUAL TO AITST IN SUBROUTINE TISUB.
J_{BB}	AJBB	SLUG-SQ IN	MOTOR PITCH AXIS MOMENT OF INERTIA ABOUT CG GRAIN.
J_{BH}	AJBH	IN**4	HEAD-END SECTION RECTANGULAR MOMENT OF INERTIA ABOUT FORWARD TANGENT PLANE FOR INCREMENTAL VOLUMES. DETERMINED IN SUBROUTINE PT1AA.
J_{BHed}	AJBHED	SLUG-SQ IN	HEAD-END SECTION TOTAL RECTANGULAR MOMENT OF INERTIA ABOUT FORWARD TANGENT PLANE.
J_{BHew}	AJBHEW	SLUG-SQ IN	RECTANGULAR MOMENT OF INERTIA OF HEAD-END WITH WEB ABOUT FORWARD TANGENT PLANE.
	AJBHN	SLUG-SQ IN	INTERMEDIATE VALUE OF HEAD-END WITH WEB RECTANGULAR MOMENT OF INERTIA DETERMINED IN THE BLOCK 2A ANALYSIS IN SUBROUTINE SCTOR1.
J_{BHO}	AJBHO	SLUG-SQ IN	HEAD-END SECTION INITIAL RECTANGULAR MOMENT OF INERTIA.
J_{BN}	AJBN	IN**4	NOZZLE-END SECTION RECTANGULAR MOMENT OF INERTIA ABOUT AFT TANGENT PLANE FOR INCREMENTAL VOLUMES. DETERMINED IN SUBROUTINE PT1AA.
J_{BNoz}	AJBNOZ	SLUG-SQ IN	NOZZLE-END SECTION RECTANGULAR MOMENT OF INERTIA ABOUT AFT TANGENT PLANE.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
J _{PH}	AJPH	IN**4	HEAD-END SECTION POLAR MOMENT OF INERTIA ABOUT ROLL AXIS FOR INCREMENTAL THIN SHELLS. DETERMINED IN SUBROUTINE PT1AA.
J _{PHed}	AJPHED	SLUG-SQ IN	HEAD-END SECTION TOTAL POLAR MOMENT OF INERTIA ABOUT ROLL AXIS.
J _{PHew}	AJPHEW	SLUG-SQ IN	POLAR MOMENT OF INERTIA OF HEAD-END WITH WEB ABOUT ROLL AXIS.
	AJPHN	IN**4	INTERMEDIATE VALUE OF HEAD-END WITH WEB POLAR MOMENT OF INERTIA DETERMINED IN THE BLOCK 2A ANALYSIS IN SUBROUTINE SCTOR1.
J _{PH0}	AJPH0	SLUG-SQ IN	HEAD-END SECTION INITIAL POLAR MOMENT OF INERTIA.
J _{PN}	AJPN	IN**4	NOZZLE-END SECTION POLAR MOMENT OF INERTIA ABOUT ROLL AXIS FOR INCREMENTAL THIN SHELLS. DETERMINED IN SUBROUTINE PT1AA.
J _{PN0z}	AJPN0Z	SLUG-SQ IN	NOZZLE-END SECTION TOTAL POLAR MOMENT OF INERTIA ABOUT ROLL AXIS.
	AJPP	IN**4	INTERMEDIATE VALUE OF POLAR MOMENT OF INERTIA OF CROSS SECTION OF INCREMENTAL AREAS IN SUBROUTINE PT1AA.
	AJPX	SLUG-SQ IN	INTERMEDIATE VALUE OF POLAR MOMENT OF INERTIA OF CROSS SECTION OF INCREMENTAL AREAS IN SUBROUTINE PT1AA.
	AJSTB	SLUG-SQ IN	INITIAL VALUE OF HEAD-END WITH WEB RECTANGULAR MOMENT OF INERTIA. DETERMINED IN SUBROUTINE ACTSUB.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	AJSTP	SLUG-SQ IN	INITIAL VALUE OF HEAD-END WITH WEB POLAR MOMENT OF INERTIA. DETERMINED IN SUBROUTINE ASTSUB.
K	AK	- - -	INPUT VALUE WHICH DETERMINES DISTANCE BETWEEN PLANES FOR THE HEAD-END WEB ANALYSIS AND FOR THE END SECTION ANALYSIS.
KG1 THRU KG5	AKG(1) THRU AKG(5)	- - -	CRITICAL MASS VELOCITY (G C R) PER UNIT AREA EQUATION CONSTANTS.
K _{GY}	AKGY	SQ IN	RADIUS OF GYRATION OF CYLINDRICAL SECTION INCREMENTAL CROSS SECTIONS. DETERMINED IN SUBROUTINE PT1AA.
	AKGYHI	SQ IN	RADIUS OF GYRATION OF UPSTREAM CYLINDRICAL SECTION INCREMENTAL CROSS SECTIONS. INITIALIZED IN SUBROUTINE SEGSUB.
	AKGYP(1) THRU AKGYP(15)	IN**4	REFERENCE PLANE GEOMETRY TABLE RADIUS OF GYRATION OF CROSS SECTION (DEPENDENT VARIABLE). THIS VARIABLE APPEARS AS INPUT ONLY IF THE MOMENT OF INERTIA OPTION (KMOICG=1) HAS NOT BEEN SUPPRESSED.
	AKGYX	SQ IN	RADIUS OF GYRATION OF REFERENCE PLANE X IN SUBROUTINE SEGSUB.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	AKGY	SQ IN	RADIUS OF GYRATION OF REFERENCE PLANE Y IN SUBROUTINE SEG SUB.
KK	AKK		ADJUSTING FACTOR USED IN THE HEAD-END WEB BLOCK 2 ANALYSIS TO DETERMINE DISTANCE BETWEEN PLANES.
KR1 THRU KR39	AKR(1) THRU AKR(39)		BURNING RATE EQUATION CONSTANTS.
	AKRF3	SQ IN	INTERMEDIATE RADIUS OF GYRATION VALUE IN SUBROUTINE MNCHN3 FOR HEAD-END WITH WEB.
KRH	AKRH	- - -	HEAD-END SECTION BURN RATE COEFFICIENT. USUAL VALUE IS THE SAME AS THE CYLINDRICAL SECTION STEADY STATE BURN RATE COEFFICIENT.
KRN	AKRN	- - -	NOZZLE END SECTION BURN RATE COEFFICIENT. USUAL VALUE IS THE SAME AS THE CYLINDRICAL SECTION STEADY STATE BURN RATE COEFFICIENT.
	AKRTAU	- - -	START TRANSIENT BURN RATE COEFFICIENT TABLE DEPENDENT VARIABLE.
KSL0T	AKSL0T(I)	- - -	SLOT BURNING RATE EQUATION COEFFICIENT.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	AKRST	- - -	ANISOTROPIC PROPELLANT BURNING RATE EQUATION COEFFICIENT DURING THE START TRANSIENT AND TAIL-OFF INTERVALS.
KU1 THRU KU5	AKU(1) THRU AKU(5)	- - -	CRITICAL GAS VELOCITY (U C R) EQUATION CONSTANTS.
L_A	ALA	IN	INITIAL LENGTH OF SECTOR 2. DETERMINED IN SUBROUTINE PLNCNS (FIGURE 5.24).
	ALA(1) THRU ALA(18)	IN	PERIMETER LENGTH OF SECTOR 1A FOR A REFERENCE PLANE.
λ	ALAMDA	RADIANS	ANGLE BETWEEN MOTOR AXIS AND A LINE FROM CENTER OF TORUS GENERATING CIRCLE TO OUTSIDE ELEMENT OF SURFACE INCREMENT ON TOROIDAL END AREA IN ZONE B OF END SECTION STRAIGHT THROUGH GRAIN. (FIGURE 5.23)
λ_{min}	ALAMIN	RADIANS	ANGLE BETWEEN MOTOR AXIS AND A LINE FROM CENTER OF TORUS GENERATING CIRCLE TO INSIDE ELEMENT OF SURFACE INCREMENT ON TOROIDAL END AREA IN ZONE B OF END SECTION STRAIGHT THROUGH GRAIN. (FIGURE 5.23)
L_B	ALB		INITIAL LENGTH OF SECTOR 4. DETERMINED IN SUBROUTINE PLNCNS (FIGURE 5.24)
	ALA(1) THRU ALB(18)	IN	PERIMETER LENGTH OF SECTOR 4 FOR A REFERENCE PLANE.
L_C	ALC	IN	INITIAL LENGTH OF SECTOR 6. DETERMINED IN SUBROUTINE PLNCNS (FIGURE 5.24).

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
L_D	ALD	IN	INITIAL LENGTH OF SECTOR 10. DETERMINED IN SUBROUTINE PLNCNS (FIGURE 5.24).
L_E	ALE	IN	INITIAL LENGTH OF SECTOR 12. DETERMINED IN SUBROUTINE PLNCNS (FIGURE 5.24).
	ALE(1) THRU ALE(18)	IN	PERIMETER LENGTH OF SECTOR LE FOR A REFERENCE PLANE.
L_P	ALP	IN	TOTAL PERIMETER LENGTH OF CURRENT INCREMENT DIVIDING PLANE CROSS SECTION FOR A SEGMENT.
	ALPHI	IN	TOTAL PERIMETER LENGTH OF UPSTREAM INCREMENT DIVIDING PLANE CROSS SECTION FOR A SEGMENT.
	ALPPL(1) THRU ALPPL(18)	IN	REFERENCE PLANE GEOMETRY TABLE PORT PERIMETER (DEPENDENT VARIABLE).
L_Q	ALQ	IN	ARC LENGTH ON PSUEDOELLIPSOID. DETERMINED IN SUBROUTINE SCTOR1 (FIGURE 5.4)
L_R	ALR	IN	LENGTH TO A POINT ON BURNING SURFACE PERIMETER IN ANY SECTOR AT THICKNESS τ MEASURED ALONG PERIMETER FROM END OF SECTOR NEAREST MOTOR AXIS. DETERMINED IN SUBROUTINE ALRSUB. (FIGURE 5.22)
L_{Rmax}	ALRMAX	IN	LENGTH OF PART OF BURNING SURFACE PERIMETER IN ANY ZONE AT THICKNESS τ MEASURED FROM END OF SECTOR NEAREST MOTOR AXIS TO END OF SECTOR OR ZONE, WHICH EVER IS SMALLER. DETERMINED IN SUBROUTINE AFSUB. (FIGURE 5.21).

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
L_{RI}	ALRI	IN	LENGTH TO THE LOWEST POINT ON BURNING SURFACE PERIMETER IN ANY ZONE AT THICKNESS τ . MEASURED ALONG PERIMETER FROM END OF SECTOR NEAREST MOTOR AXIS DETERMINED IN SUBROUTINE AESUB (FIGURE 5.21)
L_{RS}	ALRS	IN	LENGTH OF CHORD DETERMINED IN SUBROUTINE ARSSUB. (FIGURE 5.22).
L_X	ALSUBX	IN	LENGTH OF A SECTOR AT THICKNESS τ DETERMINED IN SUBROUTINE AFPSUB (FIGURE 5.21).
L_{TA}	ALTA	IN	ARC LENGTH BETWEEN POINTS PRA AND PSA (FIGURE 5.2)
L_{TB}	ALTB	IN	ARC LENGTH BETWEEN POINTS PRB AND PSB (FIGURE 5.2)
LS1A THRU LS1K	ALS1(1) THRU ALS1(18)	IN	LENGTH FROM WEB TO INNER GRAIN POINT OF REFERENCE PLANE.
LS2A THRU LS2K	ALS2(1) THRU ALS2(18)	IN	LENGTH FROM WEB TO OUTER GRAIN POINT OF REFERENCE PLANE.
AL_7	AL7	IN	PERIMETER LENGTH OF ANISOTROPIC PROPELLANT IN SECTOR 7 DURING TAIL-OFF.
AL_8	AL8	IN	PERIMETER LENGTH OF ANISOTROPIC PROPELLANT IN SECTOR 8 DURING TAIL-OFF.
MW	AMW	- - -	PRESENT TIME VALUE OF PROPELLANT GAS MOLECULAR WEIGHT.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	AMWG	LBM/MOLE	TABULAR INPUT VALUE OF PROPELLANT GAS MOLECULAR WEIGHT.
NITW	ANITW	- - -	NUMBER OF INCREMENT DIVIDING PLANES FOR WHICH WEB BURN THROUGH MUST OCCUR BEFORE TAILOFF BEGINS.
NN	ANN	- - -	NUMBER OF NOZZLES.
	ANGLE	RADIANS	CENTRAL ANGLE THAT DEFINES INTERSECTION OF ISOTROPIC PROPELLANT ARC LENGTH WITH CASE WALL IN SECTOR 7 DURING MOTOR TAIL-OFF.
	ANI	- - -	FLOATING POINT NUMBER OF TOTAL INCREMENT DIVIDING PLANES.
NOA THRU NOK	ANO(1) THRU ANO(18)	- - -	REFERENCE PLANE NUMBER OF GRAIN CROSS SECTION SYMMETRICAL PARTS.
N2	AN2	DEGREES	NOZZLE EXPANSION CONE HALF-ANGLE.
α_{01}	A01(1) THRU A01(18)	DEGREES	REFERENCE PLANE ANGLE OF SLOPE LAA WHICH DEFINES INITIAL GRAIN GEOMETRY.
α_{02}	A02(1) THRU A02(18)	DEGREES	REFERENCE PLANE ANGLE OF SLOPE LBA WHICH DEFINES INITIAL GRAIN GEOMETRY.
α_{03}	A03(1) THRU A03(18)	DEGREES	REFERENCE PLANE ANGLE OF SLOPE LCA WHICH DEFINES INITIAL GRAIN GEOMETRY.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
α_{04}	A04(1) THRU A04(18)	DEGREES	REFERENCE PLANE ANGLE OF SLOPE LDA WHICH DEFINES INITIAL GRAIN GEOMETRY.
α_{05}	A05(1) THRU A05(18)	DEGREES	REFERENCE PLANE ANGLE OF SLOPE LEA WHICH DEFINES INITIAL GRAIN GEOMETRY.
A0E	A0E	IN	LENGTH OF SEMI-MAJOR AXIS OF INNER ELLIPSE (FIGURE 5.7)
α_{0Hmax}	A0HM	DEGREES	ANGLE BETWEEN TANGENT TO FORE-HEAD ELLIPSE AND MOTOR CENTERLINE.
α_{0Nmax}	A0NM	DEGREES	ANGLE BETWEEN TANGENT TO AFT-HEAD ELLIPSE AND MOTOR CENTERLINE.
A_p	AP	SQ IN	PORT AREA OF CURRENT INCREMENT DIVIDING PLANE FOR A SEGMENT.
	APHI	SQ IN	PORT AREA OF UPSTREAM INCREMENT DIVIDING PLANE FOR A SEGMENT.
	APORT(1) THRU APORT(18)		INITIAL PORT AREA OF A REFERENCE PLANE.
A_R	AR	SQ IN	AREA OF SECTOR BCD IN ZONE A OF THE AFT END BURNING SURFACE (FIGURE 5.22)
α_{rc0}	ARCO	RADIANS	ANGLE BETWEEN Y-AXIS AND NORMAL LINE THROUGH P0 (FIGURE 5.4). USED IN SUBROUTINE SCTOR1 FOR THE HEAD-END WITH WEB B1 2A ANALYSIS

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
α_{rc1}	ARC1	RADIANS	ANGLE BETWEEN Y-AXIS AND NORMAL LINE THROUGH P1 (FIGURE 5.4). USED IN SUBROUTINE SCTORI FOR THE HEAD-END WITH WEB BLOCK 2A ANALYSIS
A_{RO}	ARO	SQ IN	AREA OF SECTOR AEF IN ZONE A OF THE AFT END BURNING SURFACE (FIGURE 5.22)
	ASE	SQ IN	END SECTION BURN AREA. COMPUTED IN SUBROUTINE ASESUB.
	ASI	SQ IN	BURNING SURFACE AREA OF AN INCREMENTAL STRIP DETERMINED IN THE BLOCK 2A ANALYSIS OF THE HEAD END WITH WEB IN SUBROUTINE SCI (FIGURE 5.4)
A_{Slvr}	ASLVR	SQ IN	INCREMENT DIVIDING PLANE INERT SLIVER AREA FOR ONE GRAIN CROSS SECTION SYMMETRICAL PART.
A_T	AT	SQ IN	EXHAUST NOZZLE THROAT AREA.
A_{TO}	ATO	SQ IN	AREA IN ZONE A OF THE AFT END BURNING SURFACE BETWEEN THE CHORD LRS AND THE CIRCULAR ARC LR (FIGURE 5.22)
A_{TT}	ATT	SQ IN	THE POSITIVE OR NEGATIVE VALUE OF ATO DEPENDING ON RT. NEGATIVE IF RT IS NEGATIVE AND POSITIVE IF RT IS POSITIVE.
	AWE	SQ IN	TOTAL BURNING SURFACE AREA OF WEB ZONE (SECTOR 8 PLUS AFT FACE) OF THE AFT END BURNING SURFACE AT THICKNESS TAU DETERMINED IN SUBROUTINE AWESUB.

ORIGINAL PAGE IS
OF POOR QUALITY

MATH SYMBOL	PROGRAM SYMROL	UNIT	DESCRIPTION
BE	BE	IN	LENGTH OF SEMI-MINOR AXIS OF END SECTION. FORE-HEAD (H) OR AFT-HEAD (N).
β_H	BH	- - -	FORE-HEAD CASE ELLIPSE RATIO.
β_N	BN	- - -	AFT-HEAD ELLIPSE RATIO.
BOE	BOE	IN	LENGTH OF SEMI-MINOR AXIS OF INNER ELLIPSE OF HEAD-END WITH WEB.
	BRNOUT	- - -	PROGRAM CONTROL FLAG FOR WEB BURNOUT.
β_{OE}	BTAOE	- - -	RATIO OF HEAD END WEB ELLIPSE AXIS NORMAL TO MOTOR AXIS TO ELLIPSE AXIS PARALLEL WITH MOTOR AXIS.
	BVX	RADIANS	ANGLE OF INERT SLIVER IN SECTOR 7.
	BVXX	RADIANS	ANGLE OF INERT SLIVER IN SECTOR 6.
	BV2	RADIANS	ANGLE THAT DEFINES PERIMETER LENGTH OF PROPELLANT IN SECTOR 7.
	BV2P	RADIANS	ANGLE THAT DEFINES PERIMETER LENGTH OF ISOTROPIC PROPELLANT ONLY IN SECTOR 7.
β_{71max}	B71M	RADIANS	GEOMETRY CONSTANT DETERMINED IN SUBROUTINE PLNCNS (FIGURE 5.25)
β_{72max}	B72M	RADIANS	GEOMETRY CONSTANT DETERMINED IN SUBROUTINE PLNCNS (FIGURE 5.25)

REV LTR A

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
CRP	CRP	- - -	CONVERGENCE VALUE FOR NON-STEADY FLOW DISCHARGE PRESSURE AT EXIT OF EACH MASS ADDITION REGION.
CRT	CRT	- - -	CONVERGENCE VALUE FOR NON-STEADY FLOW DISCHARGE GAS TEMPERATURE AT EXIT OF EACH MASS ADDITION REGION.
CRW	CRW	- - -	CONVERGENCE VALUE FOR NON-STEADY FLOW TO COMPARE FLOW RATES AT EXIT OF GRAIN TO THAT WHICH CAN BE DISCHARGED THROUGH NOZZLE BOTH AT SAME TOTAL PRESSURE.
C*	CSTAR	FT/SEC	PROPELLANT GAS CHARACTERISTICS VELOCITY.

DE-125286-1 R
224.1

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
$\beta_{91\max}$	B91M	RADIANS	GEOMETRY CONSTANT DETERMINED IN SUBROUTINE PLNCNS (FIGURE 5.25)
$\beta_{92\max}$	B92M	RADIANS	GEOMETRY CONSTANT DETERMINED IN SUBROUTINE PLNCNS (FIGURE 5.25)
	CAE CBE CCCE CCVE CDCE CDVE CFCE CEVE	- - -	CONSTANTS USED IN CALCULATION OF COEFFICIENT FOR FOURTH-DEGREE EQUATION FOR SOLUTION OF INTERSECTION OF TOROIDAL SURFACE AND END SECTION ELLIPSE. DETERMINED IN SUBROUTINE ENDCSB.
c_{F0}	CFO	- - -	THRUST COEFFICIENT DETERMINED FROM MOMENTUM EXCHANGE ONLY WITHOUT EXPANSION FROM EXIT PRESSURE TO AMBIENT PRESSURE.
	CKDUMP(1)	SEC	LOWER TIME LIMIT FOR A DIAGNOSTIC PRINT IN SUBROUTINE MNCHN4
	CKDUMP(2)	SEC	UPPER TIME LIMIT FOR DIAGNOSTIC PRINT IN SUBROUTINE MNCHN4.
	CM	- - -	NOZZLE EFFICIENCY.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	CSTR	FT/SEC	TABULAR INPUT VALUE OF CHARACTERISTIC VELOCITY FOR PROPELLANT GAS (DEPENDENT VARIABLE).
D_E	DE	IN	NOZZLE EXIT DIAMETER.
ρ_f	DELF	LB/CU IN	PROPELLANT DENSITY.
ΔL	DELL	IN	INCREMENTAL PERIMETER LENGTH USED IN APPROXIMATE INTEGRATION OF SURFACE AREAS (FIGURE 5.23)
ΔL_o	DELLO	IN	DISTANCE BETWEEN POINTS P0 AND P1 USED TO OBTAIN SURFACE AREA OF BLOCK 1 ANALYSIS OF HEAD-END WITH NO.
ΔR	DELR	IN	DISTANCE BETWEEN POINTS P0 AND P1 USED TO OBTAIN SURFACE AREA OF BLOCK 2 B AND VOLUME IN BLOCK 3 OF HEAD-END WITH WEB.
Δt	DELT	SEC	CURRENT VALUE OF TIME INCREMENT
ρ	DELTA	LB/CU IN	PROPELLANT GAS DENSITY.
Δt_{SS}	DELTSS	SEC	FIXED TIME INCREMENT FOR STEADY-STATE.
Δt_{ST}	DELTST	SEC	FIXED TIME INCREMENT USED DURING START TRANSIENT.
Δt_{TO}	DELTO	SEC	FIXED TIME INCREMENT FOR TAIL-OFF OR SHUTDOWN.

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
ΔZ	DELZ	IN	MAXIMUM LENGTH OF MASS ADDITION REGIONS IN GRAIN SEGMENTS. AT TIME=0, AN INCREMENT DIVIDING PLANE WILL BE PLACED EVERY DELZ INCHES DOWN EACH GRAIN SEGMENT BEGINNING AT THE FORWARD TANGENT PLANE OR AT THE AFT FACE OF A SLOT AND TERMINATING AT THE AFT TANGENT PLANE OR THE FORWARD FACE OF A SLOT.
D_{E1}	DE1	IN	CASE OPENING DIAMETER (H) FORE-HEAD OR (N) AFT-HEAD.
D_{H1}	DH1	IN	FORE-HEAD CASE OPENING DIAMETER.
$\frac{\Delta L}{R_f}$	DLRF	- - -	DELTA-L OVER R_f , WHERE DELTA-L = INCREMENT SIZE USED FOR THE BURNING SURFACE AREA CALCULATIONS, MEASURED ALONG THE INTERNAL PERIMETER IN THE ADJACENT REFERENCE PLANE.
D_{N1}	DN1	IN	AFT-HEAD CASE OPENING DIAMETER.
	DO3	IN	DISTANCE BETWEEN POINTS P0 AND P3. DETERMINED IN SUBROUTINE SCI FOR HEAD-END WITH WEB ANALYSIS
ΔP_o	DPO	PSIA	TOTAL PRESSURE LOSS IN NOZZLE END SECTION FROM MASS ADDITION.
	DPR	IN	DISTANCE BETWEEN POINTS POA AND POB. DETERMINED IN SUBROUTINE SCI FOR HEAD-END WITH WEB ANALYSIS.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	DPS	IN	DISTANCE BETWEEN POINTS POA AND POB. DETERMINED IN SUBROUTINE SCI FOR HEAD-END WITH WEB ANALYSIS
$\frac{\Delta R_v}{R_f}$	DRVRF -		DELTA-RV OVER RF, WHERE DELTA-RV = OUTSIDE PROPELLANT RADIUS OF ADJACENT REFERENCE PLANE.
D_T	DT	IN	INITIAL NOZZLE THROAT DIAMETER.
	DS	IN	DISTANCE BETWEEN POINTS P0 AND P1 WHEN P1 IS LOCATED ON THE CASE ELLIPSE (FIGURE 5.6-2) DETERMINED IN SUBROUTINE S2SK FOR HEAD-END WITH WEB ANALYSIS.
$\Delta \tau$	DTAU(1) THRU DTAU(18)	IN	INCREMENTAL SIZE OF REFERENCE PLANE DISTANCE BURNED TO EVALUATE GEOMETRY TABLES.
$\Delta \tau_w$	DTAUW(1) THRU DTAUW(18)		INCREMENTAL SIZE OF REFERENCE PLANE DISTANCE BURNED BEYOND WEB THICKNESS TAUW TO EVALUATE GEOMETRY TABLES.
ΔV	DV	CU IN	INCREMENTAL PROPELLANT VOLUME USED TO DETERMINE INITIAL PROPELLANT VOLUME IN END SECTIONS IN SUBROUTINE ASESUB.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
$\frac{dV}{dt}$	DVDT	CU IN/SEC	RATE OF CHANGE OF SLOT VOLUME WITH RESPECT TO TIME. COMPUTED IN SR SEG SUB FROM TSLOT AND RSLOT.
\dot{dW}	DWDOT	LB/SEC	PROPELLANT GAS GENERATED IN SEGMENT OR MASS ADDITION REGION.
$\frac{dW}{dt}$	DWDT	LB/SEC	STORED PROPELLANT GAS IN SEGMENT OR CONTROL VOLUME.
$\frac{dW_{slot}}{dt}$	DWDTS	LB/SEC	MASS OF STORED GAS IN A SLOT FOR ONE COMPUTING INTERVAL.
\dot{dW}_s	DWSLOT	LB/SEC	MASS FLOW GENERATED IN A SLOT.
ϵ_{CA}	EPCA	- - -	MEASURED CASE STRAIN AT FORWARD TANGENT PLANE.
ϵ_{CN}	EPCN	- - -	MEASURED CASE STRAIN AT AFT TANGENT PLANE.
η_1	ETA1	RADIANS	ANGLE FROM R5 RADIUS POINT THAT DEFINES INTERSECTION OF ANISOTROPIC PROPELLANT WITH CASE WALL IN SECTOR 7 DURING MOTOR TAIL-OFF.
η_2	ETA2	RADIANS	ANGLE FROM R5 RADIUS POINT THAT DEFINES INTERSECTION OF ANISOTROPIC PROPELLANT WITH ISOTROPIC PROPELLANT IN SECTOR 7 DURING MOTOR TAIL-OFF.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
η_{11}	ETA11	RADIANS	CENTRAL ANGLE THAT DEFINES INTERSECTION OF ANISOTROPIC PROPELLANT WITH CASE WALL IN SECTOR 7 DURING MOTOR TAIL-OFF.
η_{22}	ETA22	RADIANS	CENTRAL ANGLE THAT DEFINES INTERSECTION OF ANISOTROPIC PROPELLANT WITH ISOTROPIC PROPELLANT IN SECTOR 7 DURING MOTOR TAIL-OFF.
γ	GAMA	- - -	PROPELLANT GAS SPECIFIC HEAT RATIO.
γ_G	GAMAG	- - -	TABULAR INPUT VALUE OF PROPELLANT GAS SPECIFIC HEAT RATIO (DEPENDENT VARIABLE).
γ_R	GAMAR	RADIANS	ANGLE BETWEEN RADIAL VECTOR RA AND BISECTOR OF PRIMARY OR SECONDARY PROPELLANT TIP. (FIGURE 5.23)
γ_{R0}	GAMAR0	RADIANS	ANGLE BETWEEN RADIAL VECTOR RAO AND BISECTOR OF PRIMARY OR SECONDARY PROPELLANT TIP. (FIGURE 5.22)
γ_T	GAMAT	RADIANS	ANGLE SUBTENDED AT CENTER OF RADIUS RT BY CHORD LRS (FIGURE 5.20 AND 5.21).
γ_1	GAMA1	RADIANS	ANGLE BETWEEN NORMAL LINE TO PERIMETER OF GRAIN CONFIGURATION AND NORMAL LINE TO THE LINE SEGMENT RAT (FIGURE 5.6) DETERMINED IN SUBROUTINE GAMSUB.
γ_2	GAMA2	RADIANS	ANGLE BETWEEN Y-AXIS AND NORMAL LINE TO INNER ELLIPSE (FIGURE 5.7) DETERMINED IN SUBROUTINE GAMA2S.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
g_o	GNOT	FT/SQ SEC	GRAVITATIONAL CONSTANT
H_{CO}	HCO	IN	LENGTH OF CYLINDRICAL SECTION.
h_E	HE	IN	LENGTH OF A GENERAL ELEMENT OR LENGTH OF LONGER EDGE OF ELEMENT USED IN CALCULATION OF BURNING SURFACE AREA FOR THE END SECTION ANALYSIS. DETERMINED IN SUBROUTINE HESUB.
h_{EO}	HEO	IN	GEOMETRICAL LENGTH OF END SECTION. DETERMINED IN SUBROUTINE ENDCSB. (FIGURE 5.35)
h_{ER}	HER	IN	REFERENCE LENGTH OF END SECTION. DETERMINED IN SUBROUTINE ENDCSB. (FIGURE 5.35)
h_{EI}	HE1	IN	LENGTH OF END SECTION CONIC SECTION. DETERMINED IN SUBROUTINE ENDCSB (FIGURE 5.35)
h_{E2}	HE2	IN	LENGTH OF END SECTION ELLIPTIC SECTION. DETERMINED IN SUBROUTINE ENDCSB (FIGURE 5.35)
	HOLDR	IN	PREVIOUS ITERATIVE VALUE OF PERIMETER LENGTH ALONG CURRENT SECTOR. INITIALIZED IN SUBROUTINE SCI FOR THE BLOCK 1 ANALYSIS OF THE HEAD-END WITH WEB.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
		-	
	IFLAG	- - -	CONTROL FLAG IN SUBROUTINE AIBST TO INDICATE ITERATION PASSES FOR FALSE POSITION ITERATION.
	IIS	- - -	CURRENT SLOT NUMBER. USED AS A SUBSCRIPT FOR THE SLOT VARIABLES.
	IIZ	- - -	SUBSCRIPT OF RBZ TO INDICATE DIVIDING PLANE LOCATION.
	IS1	- - -	NUMBER OF INCREMENT DIVIDING PLANE THAT IS LOCATED JUST UPSTREAM OF SLOT FORWARD INTERFACE. USED IN SR SEG SUB TO OBTAIN STATIC PRESSURE AT SLOT INLET.
	IS2	- - -	NUMBER OF INCREMENT DIVIDING PLANE THAT IS LOCATED JUST DOWNSTREAM OF SLOT AFT INTERFACE. USED IN SR SEG SUB TO STORE SLOT DISCHARGE STATIC PRESSURE FOR NEXT GRAIN SEGMENT.
	KMOICG	- - -	MOMENT OF INERTIA CALCULATION SUPPRESSION FLAG. A NON-ZERO VALUE WILL SUPPRESS THE MOI AND CG CALCULATION.
	NACCEL	- - -	NUMBER OF TABLE INPUT VALUES OF ACCELERATION -TIME CURVE.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	NAKRST	- - -	NUMBER OF TABLE INPUT VALUES OF ANISOTROPIC PROPELLANT BURN RATE COEFFICIENT TABLE.
	NCSTR	- - -	NUMBER OF TABLE INPUT VALUES OF PROPELLANT GAS PROPERTIES FOR CSTR, GAMAG, AMWG, TCOMB, AND PRESS TABLES.
	NEPS	- - -	NUMBER OF TABLE INPUT VALUES OF MEASURED CASE STRAIN DATA TABLE.
	NGEOHD	- - -	NUMBER OF FORE-HEAD GEOMETRY TABLE VALUES.
	NGEO(1) THRU NGEO(18)		NUMBER OF REFERENCE PLANE GEOMETRY TABLE VALUES.
	NGEOMN	- - -	NUMBER OF AFT-HEAD GEOMETRY TABLE VALUES. 3.
	NI	- - -	INTEGER NUMBER OF TOTAL INCREMENT DIVIDING PLANES.
	NINCPL	- - -	DESIRED INCREMENT DIVIDING PLANE NUMBER TO DETERMINE BURN RATE COEFFICIENT TABLE INDEPENDENT VARIABLE, TAUAKR.
	NPH	- - -	NUMBER OF TABLE INPUT VALUES OF HEAD END PRESSURE TABLE.
	NSLOT	- - -	INTEGER NUMBER OF TOTAL SLOTS.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	NTME	- - -	NUMBER OF COMPUTED VALUES OF ANISOTROPIC PROPELLANT BURN RATE COEFFICIENT TABLE.
P_a	PA	PSIA	ATMOSPHERIC PRESSURE.
	PCTAB	- - -	PERCENT CHANGE, RELATIVE TO 1.0, IN TOTAL BURN AREA REQUIRED TO CONVERGE SOLUTION TO DESIRED VALUE OF HEAD END PRESSURE AND BURN RATE COEFFICIENT. PROGRAM WILL CLOSE ON PCTAB AT EACH TIME INCREMENT FOR SET VALUE OF PH AND AKRST AS A FUNCTION OF TIME FROM INPUT TABLES.
P_D	PD	PSIA	PRESENT TIME DISCHARGE PRESSURE OF INCREMENT DIVIDING PLANE. COMPUTED IN SUBROUTINE AIBST.
P'_D	PDPR	PSIA	PREVIOUS TIME DISCHARGE PRESSURE OF INCREMENT DIVIDING PLANE. INITIALIZED IN SUBROUTINE MNCHN4
P_{Hi}	PHI	PSIA	INITIAL GUESS OF FORE-HEAD PRESSURE FOR FIRST TIME INCREMENT.
	PHST	PSIA	DEPENDENT VARIABLE OF HEAD-END PRESSURE CURVE FIT FOR START TRANSIENT.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
π	PI	- - -	MATHEMATICAL CONSTANT.
$\frac{\pi}{2}$	PI02	- - -	PI DIVIDED BY TWO.
	PMOIHD	SLUG-SQ IN	FORE-HEAD GEOMETRY TABLE POLAR MOMENT OF INERTIA (DEPENDENT VARIABLE).
	PMOIN	SLUG-SQ IN	AFT-HEAD GEOMETRY TABLE POLAR MOMENT OF INERTIA (DEPENDENT VARIABLE).
P_o	PO	- - -	POINT ON A PLANE USED IN THE BLOCK 1 ANALYSIS OF THE HEAD-END WITH WEB. LOCATED ON THE INNER ELLIPSE ALONG A LINE PARALLEL WITH THE Y-AXIS AND NORMAL TO THE FORWARD TANGENT PLANE AT A POINT IN A SECTOR ALONG THE GRAIN INITIAL PERIMETER. (FIGURE 5.18)
			OR
			POINT ON THE PSUEDOELLIPSOID USED IN THE BLOCK 2A, 2B, AND 3 ANALYSIS OF THE HEAD-END WITH WEB.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	PRESS	PSIA	PROPELLANT GAS PROPERTY TABLE INDEPENDENT VARIABLE.
	PRNT	- - -	STORAGE ARRAY FOR EXPANDED INCREMENT DIVIDING PLANE PRINTOUT VARIABLES.
	PRTFLG		INDICATOR WHICH CONTROLS OUTPUT OF THE INTERNAL BALLISTICS MODULE DATA. PRTFLG = 0, MOTOR BALLISTIC DATA OUTPUT ONLY. = 1, MOTOR BALLISTIC DATA AND INCREMENT DIVIDING PLANE DATA WILL BE OUTPUT. = 2, MOTOR BALLISTIC DATA AND GRAIN C.G. AND M.O.I. DATA WILL BE OUTPUT. = 3, MOTOR BALLISTIC DATA, INCREMENT DIVIDING PLANE DATA, AND GRAIN C.G. AND M.O.I. DATA WILL BE OUTPUT.
P _{sa} or P _{sb}	PSA, PSB	- - -	POINT IN PLANE A OR B DEFINING RADIAL BURNING FROM POINT P0 (FIGURE 5.2) PS IS A FUNCTION OF TAU. IF TAU IS GREATER THAN OR EQUAL TO D03, THEN THE PLANE HAS BURNED OUT. IF TAU IS GREATER THAN D01, THEN PS IS LOCATED ON THE LINE SEGMENT BETWEEN POINTS P1 AND P3. IF TAU IS LESS THAN OR EQUAL TO D01, THEN PS IS LOCATED ON THE LINE SEGMENT BETWEEN POINTS P0 AND P1.
P _{ST}	PST	PSIA	MAXIMUM START TRANSIENT PRESSURE. USED AS OPTION IN SUBROUTINE TISUB TO TERMINATE START TRANSIENT CALCULATIONS.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
-------------	----------------	------	-------------

P_1		- - -	POINT ON A PLANE USED IN THE BLOCK 1 ANALYSIS OF THE HEAD-END WITH WEB. (FIGURE 5.18) LOCATED IN THE Y-Z PLANE ALONG A LINE NORMAL TO THE GRAIN PERIMETER AT P_0 .
-------	--	-------	--

OR

POINT ON THE PSUEDOELLIPSOID USED IN THE BLOCK 2A, 2B, AND 3 ANALYSIS OF THE HEAD-END WITH WEB. LOCATED AT THE DISTANCE Δ FROM POINT P_0 .

P_2		- - -	POINT ON A PLANE USED IN THE BLOCK 1 ANALYSIS OF THE HEAD-END WITH WEB. LOCATED ON THE OUTER ELLIPSOID ALONG A LINE NORMAL TO THE INNER ELLIPSOID AT P_0 . (FIGURE 5.18)
-------	--	-------	--

P_3		- - -	POINT ON A PLANE USED IN THE BLOCK 1 ANALYSIS OF THE HEAD-END WITH WEB. LOCATED IN THE Y-Z PLANE ON THE OUTER ELLIPSOID AND IN THE PLANE FORMED BY POINTS P_0, P_1 , AND P_2 (FIGURE 5.18)
-------	--	-------	--

R		FT-LBF/LBM- R^0	PROPELLANT GAS CONSTANT.
-----	--	-------------------	--------------------------

R_{Amax}	RAMAX	IN	RADIAL DISTANCE FROM MOTOR AXIS TO EITHER- 1. OUTSIDE LIMIT OF A SECTOR IN A ZONE 2. OUTSIDE LIMIT OF AN INCREMENTAL AREA (SEE FIGURES 5.21 THRU 5.23)
------------	-------	----	---

R_{Amin}	RAMIN	IN	RADIAL DISTANCE FROM MOTOR AXIS TO EITHER- 1. INSIDE LIMIT OF A SECTOR IN A ZONE 2. INSIDE LIMIT OF AN INCREMENTAL AREA (SEE FIGURES 5.21 THRU 5.23)
------------	-------	----	--

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
R_{AO}	RAO	IN	RADIAL DISTANCE FROM MOTOR AXIS TO A POINT ON PERIMETER OF ANY SECTOR AT THICKNESS τ AND LENGTH $L=0$ (FIGURE 5.21) DETERMINED IN SUBROUTINE AFSUB.
R_{aT}	RAT	IN	RADIAL VECTOR FROM MOTOR AXIS TO A POINT IN A SECTOR FOR THE BLOCK I ANALYSIS OF THE HEAD-END WITH WEB, (FIGURES 5.19 THROUGH 5.22) DETERMINED IN SUBROUTINE RASUBB.
R_{AX}	RAX	IN	RADIAL DISTANCE FROM MOTOR AXIS TO A POINT ON PERIMETER OF ANY SECTOR AT THICKNESS τ AND LENGTH $L=L_X$ FOR THE STRAIGHT THROUGH GRAIN END SECTION ANALYSIS. (FIGURE 5.34) DETERMINED IN SUBROUTINE AESUB.
R_B	RB	IN/SEC	PROPELLANT BURNING RATE OF ISOTROPIC PROPELLANT.
	RBFLAG	- - -	BURNING RATE EQUATION CONTROL FLAG FOR AKR(2) AND AKR(36). IF RBFLAG = 0.0, AKR(2) AND AKR(36) ARE INPUT IF RBFLAG = 1.0, AKR(2) AND AKR(36) = AKRST AT STEADY STATE.
	RBHI	IN/SEC	ISOTROPIC PROPELLANT BURNING RATE OF UPSTREAM INCREMENT DIVIDING PLANE
R_{Bslot}	RRSLOT	IN/SEC	BURNING RATE IN A SLOT.
	RBZTO	IN/SEC	INCREMENT DIVIDING PLANE ANISOTROPIC BURN RATE FOR SECTOR. VALUE IS EQUAL TO SECTOR 8 ANISOTROPIC BURN RATE DURING MOTOR TAIL-OFF.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
R_{B7}	RB7	IN/SEC	ANISOTROPIC PROPELLANT BURNING RATE IN SECTOR 7 DURING MOTOR TAIL-OFF.
R_{B8}	RB8	IN/SEC	ANISOTROPIC PROPELLANT BURNING RATE IN SECTOR 8 DURING MOTOR TAIL-OFF.
R_{E1}	RE1	IN	END SECTION CASE OPENING RADIUS DETERMINED IN SUBROUTINE ENDCSB. (RN1) AFT-HEAD (RH1) FORE-HEAD (FIGURE 5.31)
R_{E2}	RE2	IN	RADIAL DISTANCE FROM MOTOR AXIS TO INTERSECTION OF ELLIPTIC AND CONIC SECTION DETERMINED IN SUBROUTINE ENDCSB. (FIGURE 5.31)
R_f	RF	IN	OUTER RADIUS OF PROPELLANT.
	RF(1) THRU RF(18)	IN	REFERENCE PLANE OUTSIDE RADIUS OF PROPELLANT.
R_{IG}	RIG	IN	IGNITER OPENING RADIUS.
	RMOIHD	SLUG-SQ IN	FORE-HEAD GEOMETRY TABLE RECTANGULAR MOMENTS OF INERTIA (DEPENDENT VARIABLE).
	RMOIN	SLUG-SQ IN	AFT-HEAD GEOMETRY TABLE RECTANGULAR MOMENT OF INERTIA (DEPENDENT VARIABLE).
ρ_1	ROE1		RADIUS OF CURVATURE OF POINT P1 DETERMINED IN SUBROUTINE ROF1SR FOR THE BLOCK 2A ANALYSIS OF HEAD-END WITH WEB.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
R ₀₃	R03	IN	RADIAL DISTANCE FROM MOTOR AXIS TO ORIGIN OF R3 FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
R ₀₅	R05	IN	RADIAL DISTANCE FROM MOTOR AXIS TO ORIGIN OF R5 FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
R ₀₇	R07	IN	RADIAL DISTANCE FROM MOTOR AXIS TO ORIGIN OF R7 FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
R ₀₉	R09	IN	RADIAL DISTANCE FROM MOTOR AXIS TO ORIGIN OF R9 FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
R ₀₁₁	R011	IN	RADIAL DISTANCE FROM MOTOR AXIS TO ORIGIN OF R11 FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
R _{p1} thru R _{p13}	RP1 THRU RP13	IN	RADIAL DISTANCES DEFINING SECTOR BOUNDARIES FOR BLOCK 2B ANALYSIS OF HEAD-END WITH WEB. DETERMINED IN SUBROUTINE SCTOR1 (FIGURE 5.27)
R _{slotA}	RSLOTA	IN	RADIUS OF CASE AT SLOT AFT INTERFACE LOCATION AT PRIOR TIME INCREMENT. USED IN SR SLOT TO OBTAIN SLOT VOLUME.
R _{slotF}	RSLOTF	IN	RADIUS OF CASE AT SLOT FORWARD INTERFACE LOCATION AT PRIOR TIME INCREMENT. USED IN SUBROUTINE SLOT TO OBTAIN SLOT VOLUME.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
R_T	RT	IN	RADIUS OF CURVATURE OF PERIMETER IN ANY ZONE AT THICKNESS τ . POSITIVE WHEN CENTER LIES INSIDE ZONE, NEGATIVE WHEN CENTER LIES OUTSIDE ZONE, AND ZERO WHEN PERIMETER HAS NO CURVATURE (FIGURES 5.20 THRU 5.22) DETERMINED IN SUBROUTINE ASESUB.
R_1	R1	IN	RADIUS OF CURVATURE FROM MOTOR AXIS TO INNER STAR POINT. (FIGURE 5.3) DETERMINED IN SUBROUTINE PLNCNS.
R_2	R2(1) THRU R2(18)	IN	RADIUS OF INNER GRAIN POINT OF REFERENCE PLANE.
R_3	R3(1) THRU R3(18)	IN	FILLET RADIUS BETWEEN SLOPES LA AND LB FOR A REFERENCE PLANE.
R_4	R4(1) THRU R4(18)	IN	FILLET RADIUS BETWEEN SLOPES LB AND LC FOR A REFERENCE PLANE.
R_5	R5(1) THRU R5(18)	IN	FILLET RADIUS BETWEEN SLOPE LC AND THE WEB FOR A REFERENCE PLANE.
R_6	R6(1) THRU R6(18)	IN	FILLET RADIUS BETWEEN THE WEB AND SLOPE LD FOR A REFERENCE PLANE.
R_7	R7(1) THRU R7(18)	IN	FILLET RADIUS BETWEEN SLOPES LD AND LE FOR A REFERENCE PLANE.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
R ₈	R8(1) THRU R8(18)	IN	FILLET RADIUS BETWEEN FORKS OF FORKED WAGON WHEEL FOR A REFERENCE PLANE.
R ₉	R9	IN	RADIUS OF CURVATURE FROM MOTOR AXIS TO OUTER STAR POINT (FIGURE 5.3) DETERMINED IN SUBROUTINE PLNCNS.
	SCUR	IN	CURRENT LOCATION OF THE SLOT INTERFACE FROM THE FORWARD TANGENT PLANE. SCUR(IIS,1) IS THE FORWARD INTERFACE AND SCUR(IIS,2) IS THE AFT INTERFACE LOCATION.
	SLTFLG	- - -	PROGRAM CONTROL FLAG TO INDICATE STATUS OF THE CURRENT SLOT WITH RESPECT TO THE CURRENT INCREMENT DIVIDING PLANE. 1 - THE CURRENT INCREMENT DIVIDING PLANE AND/OR THE Y REFERENCE PLANE ARE DOWNSTREAM OF THE CURRENT SLOT FORWARD INTERFACE. THE PROGRAM WILL SET THE WORKING INCREMENT DIVIDING PLANE TO THE SLOT FORWARD INTERFACE AND CHECK THE LOCATION OF THE Y REFERENCE PLANE. 2 - THE PROGRAM IS SEARCHING FOR THE Y REFERENCE PLANE THAT IS LOCATED DOWNSTREAM OF THE CURRENT SLOT FORWARD INTERFACE. 3,4 - THE PROGRAM IS SEARCHING FOR THE Y REFERENCE PLANE THAT IS LOCATED DOWNSTREAM OF THE CURRENT SLOT AFT INTERFACE. 5 - THE X AND Y REFERENCE PLANES HAVE BEEN UPDATED FOR THE AFT SLOT DETERMINE THE SLOT MASS BALANCE.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	STDYST	- - -	CONTROL FLAG TO INDICATE STEADY STATE.
	STFLAG	- - -	PROGRAM CONTROL FLAG FOR START TRANSIENT CALCULATIONS. IF STFLAG = 1.0, PERFORM TRANSIENT CALCULATIONS.
	SA(1) THRU SA(18)	IN	SLOT FORWARD INTERFACE LOCATION FROM FORWARD TANGENT PLANE. STORED IN SCUR(11S,1).
	SB(1) THRU SB(18)	IN	SLOT AFT INTERFACE LOCATION FROM FORWARD TANGENT PLANE. STORED IN SCUR(11S,2).
	T	DEG RANKIN	COMBUSTION GAS STATIC TEMPERATURE.
	TAUAKR	IN	ANISOTROPIC BURN RATE COEFFICIENT TABLE INDEPENDENT VARIABLE.
τ_{E0}	TAUEO	IN	MAXIMUM BURNING DISTANCE IN STRAIGHT THROUGH GRAIN END SECTION (FIGURE 5.30) DETERMINED IN SUBROUTINE ENDCSB
τ_{E1}	TAUE1	IN	MAXIMUM BURNING DISTANCE IN STRAIGHT THROUGH GRAIN END SECTION CONIC SECTION. (FIGURE 5.30) DETERMINED IN SUBROUTINE ENDCSB.
	TAUHD	IN	FORE-HEAD GEOMETRY TABLE INDEPENDENT VARIABLE (DISTANCE BURNED).

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
τ_{\max}	TAUM	IN	MAXIMUM BURNING DISTANCE FOR A REFERENCE PLANE. DETERMINED IN SUBROUTINE ENDCSB FROM THE GEOMETRY PLANE CONSTANTS.
	TAUN	IN	AFT-HEAD GEOMETRY TABLE INDEPENDENT VARIABLE (DISTANCE BURNED).
	TAUPL(1) THRU TAUPL(18)	IN	REFERENCE PLANE GEOMETRY TABLE DISTANCE BURNED (INDEPENDENT VARIABLE).
τ_{TO}	TAUTO	IN	ANISOTROPIC PROPELLANT THICKNESS. CALCULATED AT END OF START TRANSIENT IN SUBROUTINE SEGSUB. EQUAL TO TAUAKR (NTIME).
	TAUTOV	IN	TEMPORARY VALUE FOR TAUTO IN SUBROUTINES LPDAPS AND AFPSUB.
	TAUTOZ	IN	ANISOTROPIC PROPELLANT THICKNESS OVER INERT SLIVER IN SECTORS 6 AND 7.
	TAUW(1) THRU TAUW(18)	IN	REFERENCE PLANE WEB THICKNESS.
	TAUWDP	IN	INCREMENT DIVIDING PLANE WEB THICKNESS.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	TAUZTO	IN	TOTAL DISTANCE BURNED OF ANISOTROPIC PROPELLANT IN SECTORS 7 AND 8 DURING MOTOR TAIL-OFF.
	TCOMB	DEG RANKIN	TABULAR INPUT VALUE OF COMBUSTION TEMPERATURE OF PROPELLANT GAS (DEPENDENT VARIABLE).
	TDMAX	IN	MAXIMUM BURNING DISTANCE IN THE FORE-HEAD WITH A HEAD-END WEB. DETERMINED IN SUBROUTINE SCI DURING THE INITIAL INTEGRATION FOR THE BURNING SURFACE AREA IN THE BLOCK 1 ANALYSIS. EQUAL IN VALUE TO TAUMA OR D03 FOR AN INTEGRATION PLANE.
θ_o	THO	RADIANS	CENTRAL ANGLE OF SLOTTED- CONE GRAIN CONFIGURATION TO DEFINE LOCATION OF ONE-HALF OF SLOT IN ONE GRAIN CROSS SECTION SYMMETRICAL PART.
θ_{oA} thru θ_{oK}	THO(1) THRU THO(18)	DEGREES	REFERENCE PLANE SLOTTED-CONE GRAIN CONFIGURATION CENTRAL ANGLE.
θ_R	THR	RADIANS	ANGLE BETWEEN Z-AXIS AND A RADIAL VECTOR FROM MOTOR AXIS TO A POINT ON A SECTOR. DETERMINED IN SUBROUTINE THETAR (FIGURE 5.21)

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
θ_{RI}	THRI	RADIANS	ANGLE BETWEEN Z-AXIS AND A RADIAL VECTOR FROM MOTOR AXIS TO A POINT ON A SECTOR. DETERMINED IN SUBROUTINE XRTHR (FIGURE 5.28)
θ_{RO}	THRO	IN	ANGLE BETWEEN Z-AXIS AND RADIAL VECTOR RAO.
θ_{slv}	THSLV	DEGREES	HALF ANGLE OF INERT SLIVER SECTOR OF INCREMENT DIVIDING PLANE IN SUBROUTINE SEGSUB
θ_{slvA} thru θ_{slvK}	THSLV(1) THRU THSLV(18)	DEGREES	REFERENCE FRAME INERT SLIVER SECTOR HALF ANGLE.
	THSLVV	DEGREES	INCREMENT DIVIDING PLANE INERT SLIVER SECTOR HALF ANGLE IN SUBROUTINE AFPSUB.
	THSLVX AND THSLVY	DEGREES	REFERENCE FRAME SECTOR HALF ANGLE IN SUBROUTINE SEGSUB.
θ_1	TH1	RADIANS	GEOMETRY CONSTANT DETERMINED IN SUBROUTINE PLNCNS (FIGURE 5.3)
θ_2	TH2	RADIANS	GEOMETRY CONSTANT DETERMINED IN SUBROUTINE PLNCNS. (FIGURE 5.3)
θ_3	TH3	RADIANS	GEOMETRY CONSTANT DETERMINED IN SUBROUTINE PLNCNS. (FIGURE 5.3)
θ_4	TH4	RADIANS	GEOMETRY CONSTANT DETERMINED IN SUBROUTINE PLNCNS. (FIGURE 5.3)
	TIMAX	SEC	MAXIMUM VALUE OF TIME. IF TIME IS GREATER THAN OR EQUAL TO TIMAX, PROGRAM EXECUTION WILL TERMINATE FOR THE CASE.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	TIMEAC	SEC	ACCELERATION-TIME CURVE INDEPENDENT VARIABLE.
	TIMEPH	SEC	HEAD END PRESSURE CURVE INDEPENDENT VARIABLE.
	TIMEPS	SEC	MEASURED CASE STRAIN DATA TABLE INDEPENDENT VARIABLE.
T_o	TO	DEG RANKIN	PROPELLANT GAS COMBUSTION TEMPERATURE.
	TOFLAG	- - -	PROGRAM CONTROL FLAG TO SIGNAL SUBROUTINE RBSUB TO COMPUTE BURN RATE FROM ANISOTROPIC BURN RATE COEFFICIENT TABLE. SET IN SUBROUTINE SEGSUB TO COMPUTE BURN RATE OVER INERT SLIVER IN SECTORS 6 AND 7 AND SET IN SUBROUTINE MNCHN4 TO COMPUTE BURN RATE IN FORE AND AFT DOMES DURING TAIL-OFF.
	TPR	- - -	PRESSURE RATIO OF NOZZLE TOTAL PRESURE TO HEAD END TOTAL PRESSURE. USED IN SUBROUTINE RBSTSB TO DETERMINE FIRST ITERATION VALUE OF AKRST FOR EACH TIME INCREMENT.
τ_{slotA}	TSLOTA	IN	ONE PAST TIME VALUE OF LATERAL DISTANCE BURNED IN A SLOT FORWARD INTERFACE. USED IN CONJUNCTION WITH SKA TO DEFINE SLOT INTERFACE LOCATION AND TO DETERMINE DV/DT OF A SLOT.
τ_{slotF}	TSLOTF	IN	ONE PAST TIME VALUE OF LATERAL DISTANCE BURNED IN A SLOT FORWARD INTERFACE.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
	TSLVR	IN	INERT SLIVER RADIUS FROM R5 FILLET TO INERT SLIVER IN SUBROUTINE SEGSUB.
	TSLVR(1) THRU TSLVR(18)	IN	DISTANCE FROM CORE INTERFACE TO INERT SLIVER FOR A REFERENCE PLANE.
	TSLVRN	IN	SLIVER RADIUS OF REFERENCE PLANE ADJACENT TO NOZZLE SECTION. DETERMINED BY SUBROUTINE SEGSUB FOR USE IN SUBROUTINE TISUB TO LIMIT TAUZTO(NI) TO SLIVER RADIUS.
	TSLVRV	IN	INERT SLIVER RADIUS OF INCREMENT DIVIDING PLANE IN SUBROUTINE AFPSUB.
	TSLVRX	IN	INERT SLIVER RADIUS OF CURRENT REFERENCE PLANE IN SUBROUTINE SEGSUB.
	TSLVRY	IN	INERT SLIVER RADIUS OF NEXT REFERENCE PLANE IN SUBROUTINE SEGSUB.
	TST	SEC	START TIME, USED AS OPTION IN SUBROUTINE TISUB TO TERMINATE START TRANSIENT INTERVAL.
	TV6PR	IN	REFERENCE PLANE SLOTTED-CONE GRAIN CONFIGURATUON GEOMETRY CONSTANT. DISTANCE BETWEEN SIDE LC AND CASE WALL MEASURED ALONG A PERPENDICULAR LINE TO LC AT THE INTERSECTION OF LC WITH RADIUS R5.
	TV7PR	IN	REFERENCE PLANE SLOTTED-CONE GRAIN CONFIGURATION GEOMETRY CONSTANT. DISTANCE ALONG LINE TV6PR WHERE SIDE LC DISAPPEARS AS A RESULT OF PROGRESSION OF BURNING SURFACE OF SLOTTED-CONE.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
τ_{2max}	T2M	IN	MAXIMUM BURNING DISTANCE FOR A SECTOR DETERMINED IN SUBROUTINE PLNCNS (FIGURE 5.3)
.	T4M		
.	T5M		
.	T6M		
	T7M		
	T9M		
τ_{12max}	T10M		
	T12M		
	U	FT/SEC	VELOCITY OF GAS IN CONTROL VOLUME.
	V	CU IN	PRESENT TIME GAS VOLUME OF INCREMENT DIVIDING SECTION. USED IN SUBROUTINE AIBST TO COMPUTE PD.
V_{CE}	VCE	CU IN	VOLUME OF CASE IN REFERENCE END SECTION DETERMINED IN SUBROUTINE ENDCSB.
	VCHINP	CU IN	VOLUME OF FORE-HEAD CASE.
	VCNINP	CU IN	VOLUME OF AFT-HEAD CASE.
V_{EH}	VEH	CU IN	INITIAL VOLUME OF PROPELLANT BETWEEN TWO OBLATE SPHEROIDS. DETERMINED IN SUBROUTINE VOLSUB FOR THE BLOCK 3 ANALYSIS OF THE HEAD-END WITH WEB.
V_{fHO}	VFHO	CU IN	INITIAL PROPELLANT VOLUME IN FORE-HEAD.
V'_{fH}	VFHPR	CU IN	PREVIOUS TIME VALUE OF HEAD-END FUEL VOLUME.
	VFI	CU IN	VOLUME OF FUEL IN INCREMENT DIVIDING PLANE.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
V_{fno}	VFNO	CU IN	INITIAL PROPELLANT VOLUME IN AFT-HEAD.
	VOLCH	CU IN	TOTAL HEAD END VOLUME CONSUMED.
	VOLCHO	CU IN	PREVIOUS TIME VALUE OF HEAD END VOLUME CONSUMED.
	VOLCN	CU IN	TOTAL NOZZLE END VOLUME CONSUMED.
	VOLCNO	CU IN	PREVIOUS VALUE OF NOZZLE-END TOTAL VOLUME CONSUMED.
	VPH	CU IN	FREE GAS VOLUME OF HEAD-END SECTION.
	VPN	CU IN	FREE GAS VOLUME OF NOZZLE END SECTION.
V'	VPR	CU IN	PREVIOUS TIME VALUE OF INCREMENT DIVIDING SECTION GAS VOLUME.
V_{slvr}	VSLVR	CU IN	VOLUME OF INERT SLIVER.
\dot{W}_D	WDOTD	LB/SEC	TRANSIENT DISCHARGE WEIGHT FLOW FROM INCREMENT DIVIDING SECTION.
\dot{W}_{Dslot}	WDSLOT	LB/SEC	DISCHARGE MASS FLOW OF A SLOT.
\dot{W}_{Islot}	WISLOT	LB/SEC	INLET MASS FLOW OF A SLOT.
	XCGHD	IN	FORE-HEAD GEOMETRY TABLE CENTER OF GRAVITY LOCATION FROM FORWARD TANGENT PLANE (DEPENDENT VARIABLE).
	XCGN	IN	AFT-HEAD GEOMETRY TABLE CENTER OF GRAVITY LOCATION FROM AFT TANGENT PLANE (DEPENDENT VARIABLE).

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
X ₀₃	X03	IN	X-COORDINATE OF R ₃ FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
X ₀₅	X05	IN	X-COORDINATE OF R ₅ FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
X ₀₇	X07	IN	X-COORDINATE OF R ₇ FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
X ₀₉	X09	IN	X-COORDINATE OF R ₉ FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
X ₀₁₁	X011	IN	X-COORDINATE OF R ₁₁ FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
X _R	XR	IN	X-COORDINATE OF A POINT ON THE PERIMETER OF A SECTOR. DETERMINED IN SUBROUTINE XRSUR FOR THE END SECTION STRAIGHT THROUGH GRAIN ANALYSIS.
X _{ra}		IN	X-COORDINATE OF A POINT ON THE PERIMETER OF A SECTOR FOR THE BLOCK 1 ANALYSIS OF THE HEAD-END WITH WEB. DETERMINED IN SUBROUTINE XRSUBB.
X _{rl}	XRI	IN	X-COORDINATE OF A POINT ON THE PERIMETER OF A SECTOR FOR THE BLOCK 2B ANALYSIS OF THE HEAD-END WITH WEB (FIGURE 5.28) DETERMINED IN SUBROUTINE XRSUBB.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
X_{Rmin}		IN	X-COORDINATE OF A POINT ON THE PERIMETER OF ANY SECTOR AT THICKNESS TAU AND RA=RAMIN (FIGURE 5.34)
X_{R0}	XRO	IN	X-COORDINATE OF THE POINT ON THE PERIMETER OF ANY SECTOR AT THICKNESS TAU AND RA=RA0. (FIGURE 5.34)
X_{RX}		IN	X-COORDINATE OF A POINT ON THE PERIMETER OF ANY SECTOR AT THICKNESS TAU AND RA=RAX. (FIGURE 5.34)
X_{45}	X45	IN	GEOMETRY CONSTANT WHICH DEFINES THE X-COORDINATE OF THE POINT LOCATED ALONG LINE T6 MAX AT THE DISTANCE R5 FROM SIDE LC (FIGURE 5.3) DETERMINED IN SUBROUTINE PLNCNS.
X_{76}	X76	IN	GEOMETRY CONSTANT WHICH DEFINES THE X-COORDINATE OF THE POINT LOCATED ALONG LINE T12 MAX AT THE DISTANCE R6 FROM SIDE LD (FIGURE 5.3) DETERMINED IN SUBROUTINE PLNCNS.
Y_{Amax}		IN	Y-COORDINATE CORRESPONDING TO XRMAX (FIGURE 5.34)
Y_{Amin}		IN	Y-COORDINATE CORRESPONDING TO XRMIN (FIGURE 5.34)
Y_{A0}		IN	Y-COORDINATE CORRESPONDING TO XRO (FIGURE 5.34)
Y_{AX}		IN	Y-COORDINATE CORRESPONDING TO XRX (FIGURE 5.34)

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
Y _{NO}	YN0	IN	Y-INTERCEPT OF LINE NORMAL TO INNER ELLIPSE AT RADIUS RIG WHICH IS USED TO DETERMINE ANGLE BETWEEN Y-AXIS AND NORMAL LINE IN SUBROUTINE AIGSUB (FIGURE 5.26)
Y _{0A}	Y0A	IN	Y-COORDINATE OF THE P0 POINT FOR PLANE A.
Y _{0B}	Y0B	IN	Y-COORDINATE OF THE P0 POINT FOR PLANE B.
		IN	Y-COORDINATE OF ORIGIN OF CIRCULAR ARC USED IN CALCULATION OF AFT-END SECTOR AREA IN SUBROUTINE ASESUB.
Y ₀₃	Y03	IN	Y-COORDINATE OF R3 FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS
Y ₀₅	Y05	IN	Y-COORDINATE OF R5 FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
Y ₀₇	Y07	IN	Y-COORDINATE OF R7 FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
Y ₀₉	Y09	IN	Y-COORDINATE OF R9 FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
Y ₀₁₁	Y011	IN	Y-COORDINATE OF R11 FILLET RADIUS (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
Y ₄₅	Y45	IN	GEOMETRY CONSTANT WHICH DEFINES THE Y-COORDINATE OF THE POINT LOCATED ALONG LINE T6 MAX AT THE DISTANCE R5 FROM SIDE LC (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.

MATH SYMBOL	PROGRAM SYMBOL	UNIT	DESCRIPTION
Y_{76}	Y76	IN	GEOMETRY CONSTANT WHICH DEFINES THE Y-COORDINATE OF THE POINT LOCATED ALONG THE LINE T12 MAX AT THE DISTANCE R6 FROM SIDE LD (FIGURE 5.4) DETERMINED IN SUBROUTINE PLNCNS.
	ZCALC(I)	IN	STORAGE ARRAY OF INCREMENT DIVIDING PLANE. LOCATIONS.
Z_{1aT}	Z1AT	IN	(1) DISTANCE FROM CENTER OF ELLIPSE TO POINT ON OUTER ELLIPSE (FIGURE 5.19) (2) CODE USED IN SUBROUTINE P3SUB.
α_V		RADIANS	ANGLE BETWEEN BISECTOR OF PROPELLANT TIP AND STRAIGHT SIDE SECTOR USED IN CALCULATION OF AFT-END SECTOR AREA IN SUBROUTINE ASESUB.

The major modifications to the original Thiokol Chemical Corporation (TCC) program made by Boeing were done during the HIBEX program. HIBEX stands for "High G Boost Experiment." Because of its extreme flight environment and a new stapled high burning rate propellant, it was not known if the internal ballistics of the HIBEX motor would be affected, i.e., ignition transient interval, maximum chamber pressure, motor burn time, and shape of the chamber pressure-time trace. Accordingly, program modifications were made as discussed in Section 1.1.

Resulting predictions using these modifications were compared with measured results. The dimensionless fore-head pressure-time traces are shown in Figure 7.1. The prediction indicates good agreement with the results of three full scale motor firings. During the HIBEX program, various grain configurations, propellant formulations, and nozzle throat sizes were used. The three firings for which data are shown in Figure 7.1 are from identical motor configurations.

Security classification of the HIBEX program prohibits the discussion of specific numerical results in this unclassified report. Initial analysis that was conducted is reported in Reference 4, and an updated analysis using an anisotropic burning rate model developed from small scale "Forty-Pound Charge" (FPC) motor firings to predict the traces shown in Figure 7.1 is discussed in Reference 5.

Figure 7.2 shows the influence of internal gas flow along the propellant grain and the propellant burning rate model on ballistic predictions. Curve A is based on steady internal flow and isotropic burning. Curve B is based on non-steady internal flow and isotropic burning. Curve C represents use of the complete program capability except for an accelerating reference system. It is based on non-steady, internal flow and anisotropic burning during ignition and tailoff. The method of solution for these and other program options are discussed in Section 8.1.

USE FOR TYPEWRITTEN MATERIAL ONLY

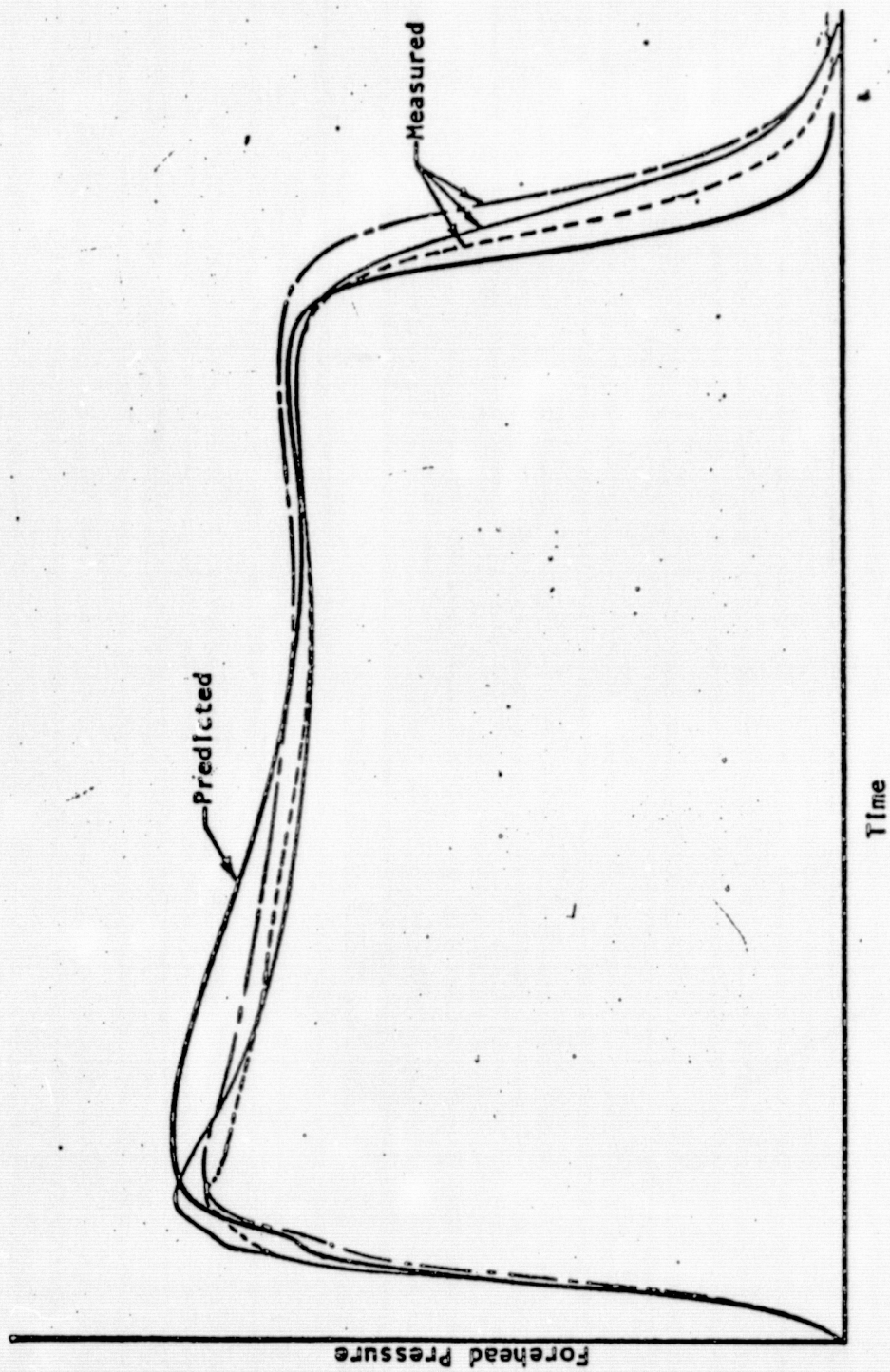


Figure 7.1. HIBEX Forehead Chamber Pressure Trace

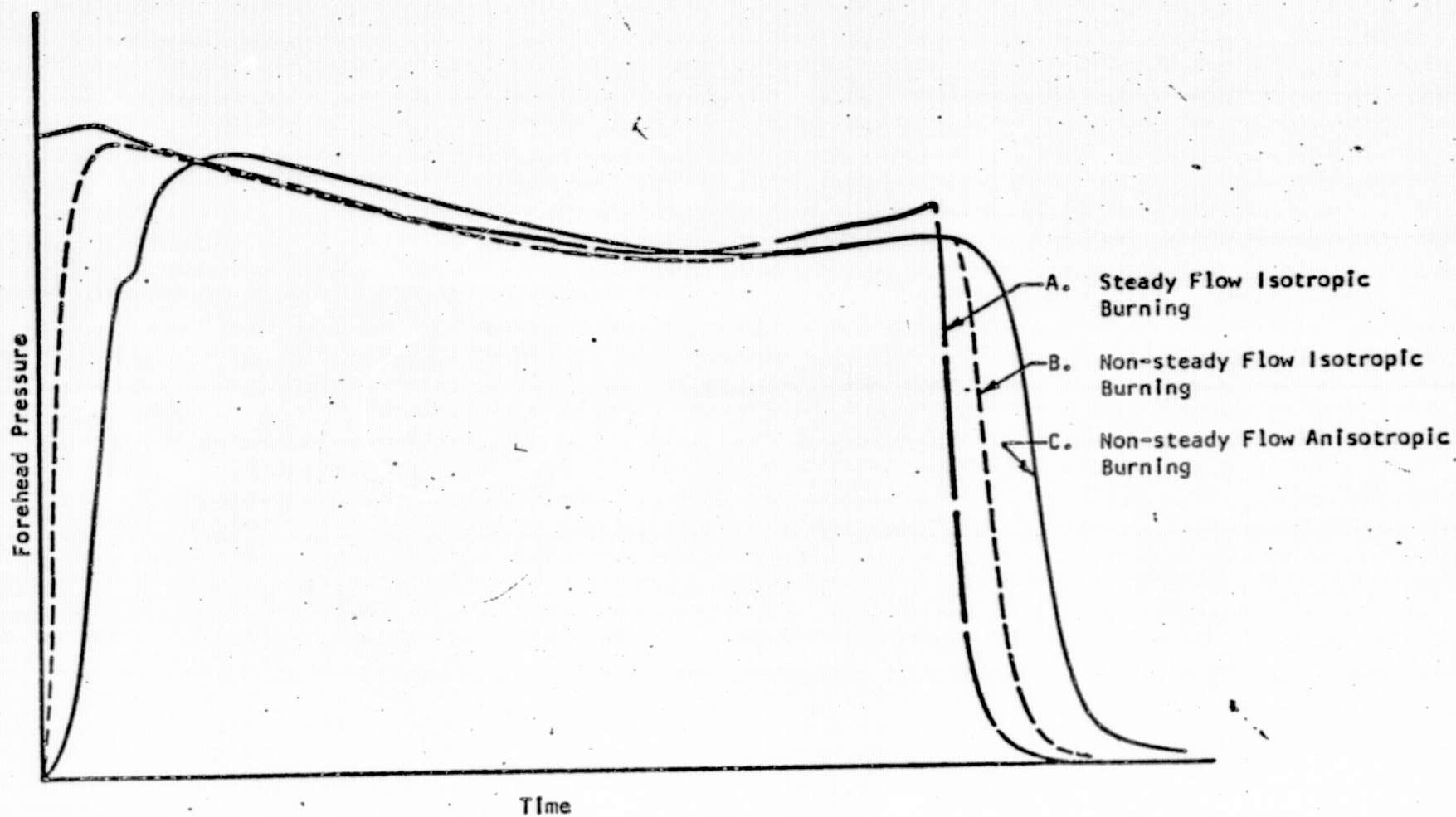


Figure 7.2. Influence of Internal Flow and Burning Rate Model